N. 30/1991
Sped. in abb. post. gr. III/70
L. 15.000
68000
AMIGA
BYTE
by Elettronica 2000
Tutorial
LE ANIMAZIONI
CON DELUXE PAINT III
Midi
TIGER CUB, THE COPYIST,
BARS & PIPES
Grafica
3D PROFESSIONAL
Linguaggi
IL COMPILATORE
FAST BASIC 2.0
Didattica
IL KICKSTART 2.0
SU A500 E A2000
TIPS & TRICKS
Dtp
PUBLISHING PARTNER MASTER
CONTRO SAXON PUBLISHER
Tools
AUDIOMASTER III
ZX SPECTRUM EMULATOR
SUL DISCO
13
PROGRAMMI
BANNERPRESS STAMPA STRISCIONI
PBAR EDITOR DI FINESTRE
INFILTRATOR MEGA RIPPER
BOUNCER ABBATTI IL MURO
WO MINI AGENDA INDIRIZZI
COLORCATCH CATTURA I COLORI
HYPERBALL LA SFERA INFERNALE
FIXDISK CORREGGE ERRORI SU DISCO
APRF STAMPA FORMATTATA TESTI
POWERMENU CREA MENU PER DISCO
VSCAN ANNIENTA OLTRE 100 VIRUS
UBINSTALL UN SUPER BOOTBLOCK
CONTOURA FUNZIONI GRAFICHE 2D
TEMPESTA

AMIGA BYTE

Direttore
SIRA ROCCHI

Direzione Editoriale
MARIO MAGRONE

Direzione Tecnica
GIANCARLO CAIRELLA

Segreteria di Redazione
SILVIA MAIER

Grafica
NADIA MARINI

Fotografie
MARIUS LOOK

Disco a cura di
VITTORIO FERRAGUTI

Copertina
FRANCO TEMPESTA

Collaborano ad AmigaByte: Luca Arienti, Francesco Annoni, Laura Baricevic, Paolo Bozzo, Luca Brigatti, Marco Brovelli, Paolo Colombo, Antonio De Lorenzo, Enrico Donna, Enrico Frascati, Renato Grossi, Fabrizio Lodi, Silvia Malaguti, Vincenzo Marangoni, Dario Martinelli, Luca Mirabelli, Lorenzo Orlandini, Domenico Pavone, Graziano Pavone, Roberto Pellagatti, Riccardo Premoli, Guido Quaroni, Fabio Rossetti, Emanuele Scribanti, Paolo Sisti, Leonardo Tennozio, Aurora Tragara, Vertigo.

Redazione
C.so Vitt. Emanuele 15
20122 Milano
tel. 02/795047
ore pomeridiane.
Per telefonate tecniche: solo mercoledì h 15-18.

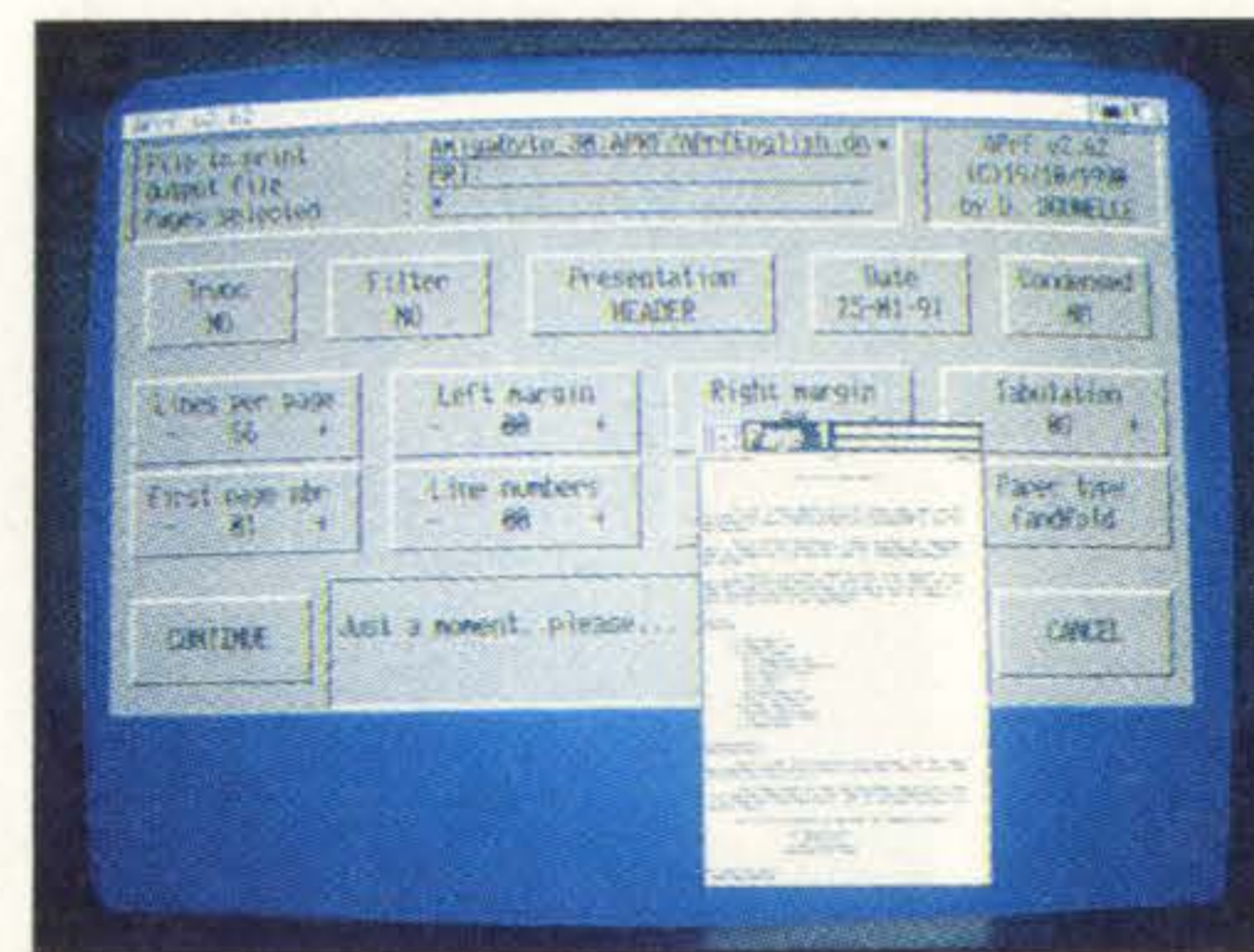
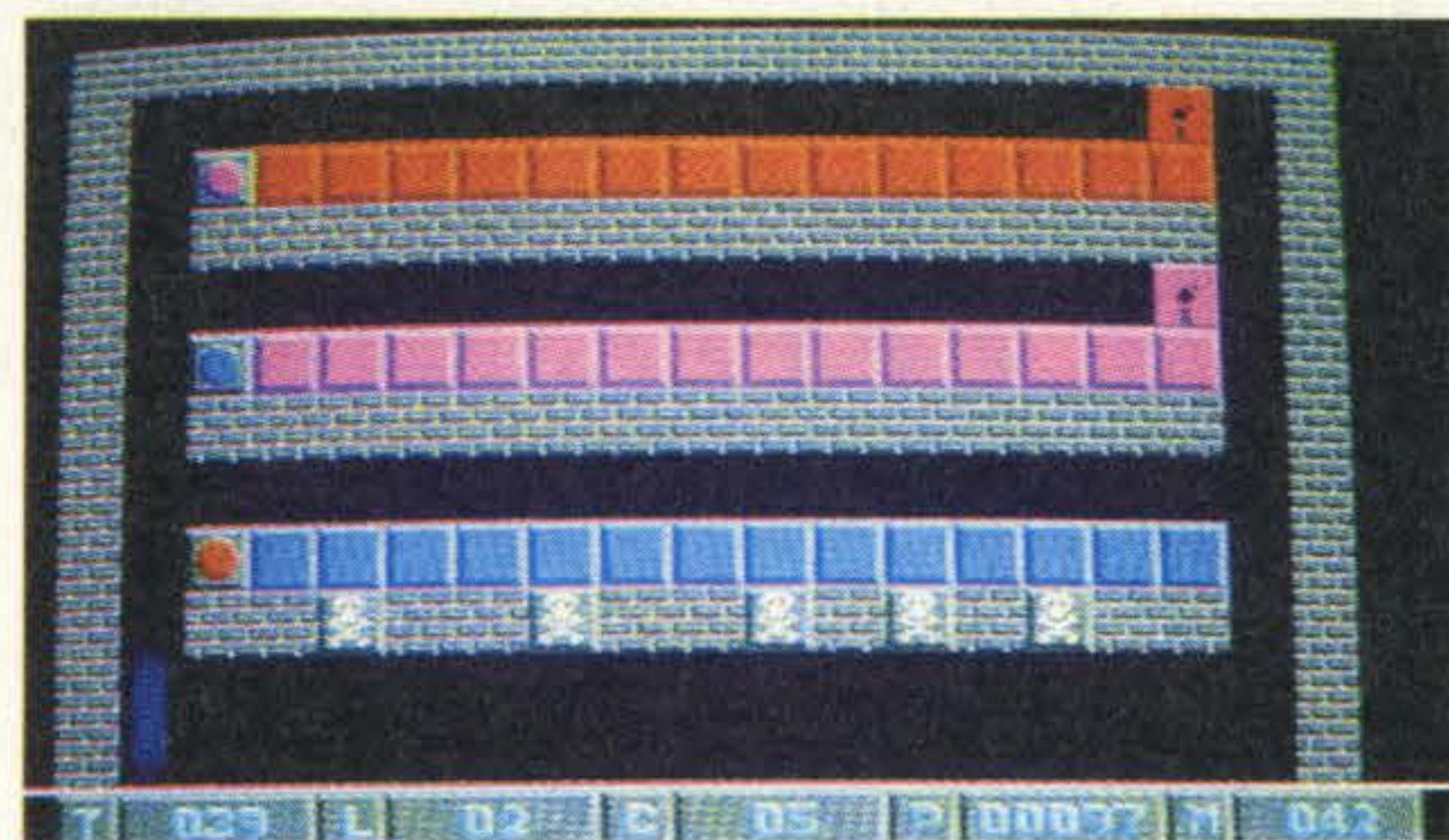
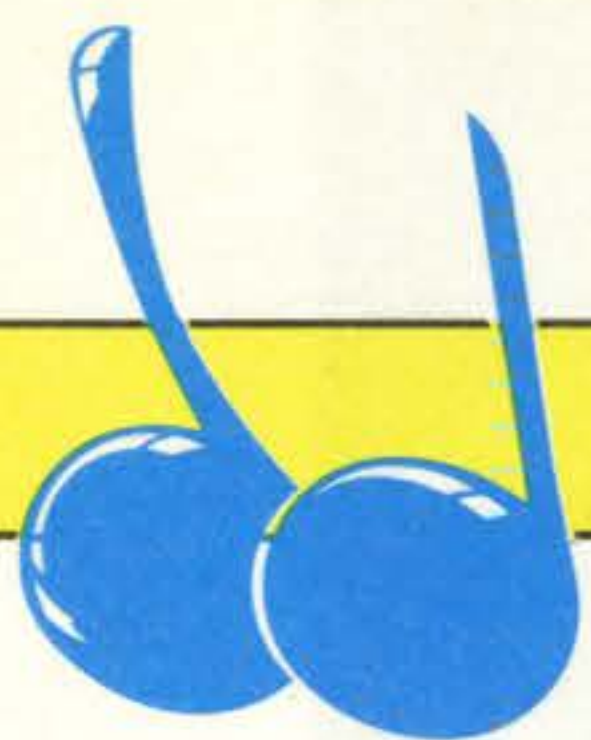
Amministrazione, Redazione, Pubblicità: L'Agorà srl: C.so Vittorio Emanuele 15, 20122 Milano. Fotocomposizione: Compostudio Est, selezioni colore e fotolito: Eurofotolit. Stampa: Garzanti Editore S.p.A. Cernusco S/N (MI). Distribuzione: SO.DI.P. Angelo Patuzzi spa, Via Zuretti 25, Milano. Amiga Byte è un periodico mensile registrato presso il Tribunale di Milano al n. 215 il 29 marzo 1988. Resp. Sira Rocchi. Spedizione in abbonamento postale Gr. III/70. Pubblicità inferiore al 70%.

IL MENU

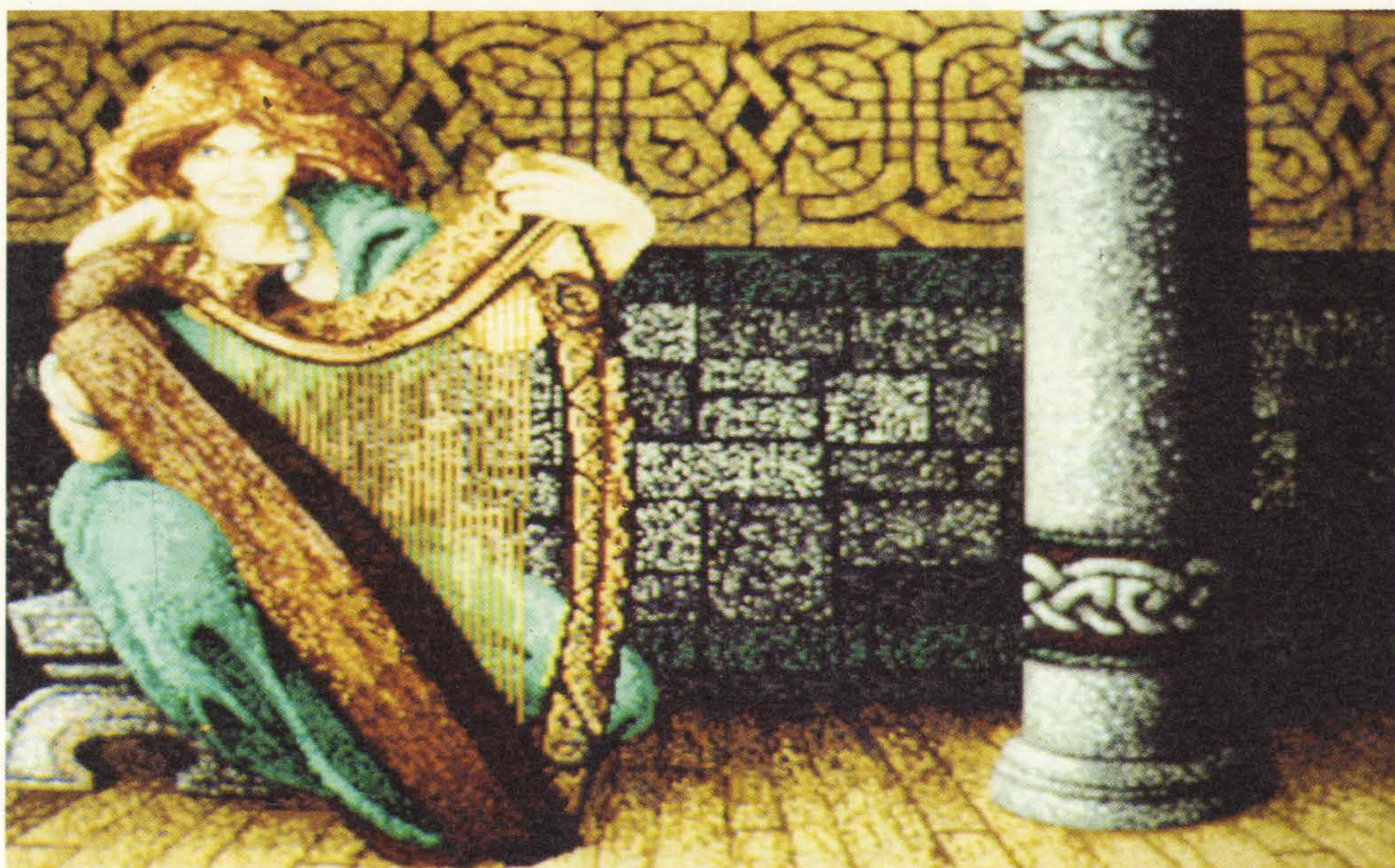
4 - MIDI
9 - LETTERE
10 - DTP A CONFRONTO
18 - KICKSTART 2.0 PER TUTTI
21 - ANIMARE CON DPAINT III
27 - 3D PROFESSIONAL
33 - FAST BASIC COMPILATORE
40 - TOOLS
44 - TIPS & TRICKS
45 - RECENSIONI GIOCHI

SUL DISCHETTO...

Come sempre il dischetto di AmigaByte presenta una collezione di programmi e giochi adatti a tutte le esigenze, da quelle degli utenti alle prime armi a quelle degli smanettoni più esperti, che vogliono mantenersi aggiornati sulle novità del panorama del software di pubblico dominio.
FIXDISK non è una novità per chi ci segue da tempo: si tratta appunto di una versione aggiornata e potenziata di un'utility apparsa originariamente sul dischetto allegato al fascicolo 16 di AmigaByte. La sua funzione consiste nell'individuare e tentare di correggere gli eventuali errori presenti sui dischetti: questa nuova versione 1.2 consente ora di effettuare le stesse operazioni di

salvataggio anche su hard disk.
Una nuova e potentissima arma decisiva nell'eterna lotta contro i virus è **VSCAN**, un formidabile antivirus in grado di riconoscere decine di virus letali dell'ultima generazione nascosti nei bootblock o annidati nei file eseguibili.
Per i possessori di stampante, due simpatici tools: **APRF**, dal nome impronunciabile, rende semplicissimo stampare testi su carta definendone il formato, la lunghezza in righe ed il tipo di impaginazione; **BANNERPRESS** produce invece striscioni su carta a modulo continuo ed è l'ideale per creare scritte di grandi dimensioni che non passano inosservate.
Dal nome non si direbbe, ma **WO** è un'agenda/database creata appositamente per memorizzare nominativi ed indirizzi; la sua peculiarità consiste nel permettere la protezione, tramite una password, dei dati in essa contenuti.
Gli utenti del WorkBench apprezzeranno **PBAR**, un editor che permette di variare l'aspetto della barra superiore delle finestre definendo motivi ornamentali e salvandoli sotto forma di file eseguibili, e **COLORCATCH**, che invece memorizza le configurazioni dei colori dello schermo permettendovi di richiamare quelle più gradite in pochi istanti e senza ricorrere a Preferences.
Quante volte vi è capitato di dover scollegare i drive esterni per mancanza di memoria o, al contrario, di dover disabilitare l'espansione per quei programmi che non girano in sua presenza? **UBINSTALL** risolve il problema, installando sui vostri dischetti un bootblock particolare che vi permetterà di selezionare questi ed altri parametri operativi immediatamente dopo il reset, senza ricorrere a NoFastMem o ad altre scomode diavolerie.
Chi desidera creare menu personalizzati sui propri dischetti per richiamare via mouse i programmi in essi contenuti può farlo con **POWERMENU**, un tool facilissimo da usare che vanta un look ed una funzionalità davvero superlativi.
INFILTRATOR è un'altra utility per hacker incalliti, nella tradizione dei «ripper» precedentemente apparsi su AmigaByte: permette di caricare ed esaminare fin nei minimi dettagli qualsiasi file, alla

ricerca di dati o stringhe di testo da modificare. Il suo algoritmo di ricerca sconfigge parecchi metodi di crittazione dei testi.
Amate la grafica? Con **CONTOURA** potrete dedicarvi al tracciamento di disegni coloratissimi basati su funzioni matematiche, e realizzare suggestive immagini Iff.
Infine, gli immancabili giochi del mese, accomunati questa volta da un elemento in comune: le palle. Prima che pensiate male, precisiamo che **BOUNCER** è una versione frenetica del classico Arkanoid, mentre **HYPERBALL** è un gioco arcade che presenta elementi di strategia, nel quale occorre eliminare dei blocchi facendoli urtare contro una sfera dello stesso colore.

Tiger Cub, The Copyist, Bars & Pipes

Una panoramica su tre interessanti pacchetti software di gestione Midi, decisamente insoliti nell'impostazione e nelle funzionalità

di SIMONE CORELLI ed ENRICO DONNA

Dopo la prova di «**Music-X**» nel fascicolo di dicembre 1989, AmigaByte si occupa nuovamente di software musicale di sequencing e, più in generale, di software per la gestione di strumenti tramite l'interfaccia seriale **Midi**.

Da un po' di tempo, il settore della computer music inizia a muoversi anche su Amiga. Finalmente cominciano ad essere disponibili anche per il nostro computer alcuni dei programmi che la software house **Dr. T's Music Software** aveva sviluppato per Atari ST e Macintosh: oltre che il famoso «**Keyboard Controlled Sequencer**» (più noto come «KCS»), sono da qualche tempo in commercio programmi come «**Tiger Cub**» e lo splendido «**The Copyist Professional**», il *word processor* musicale.

C'è dunque di che essere soddisfatti: dopo una lunga attesa, il livello di diffusione di Amiga ha convinto le software house specializzate in prodotti musicali a dedicare un poco di attenzione anche a questo computer, finora considerato con poco riguardo nell'ambiente della «computer music» professionale. Nell'attesa di veder apparire anche sugli schermi di Amiga programmi del calibro del leggendario «**Notator**», ecco una breve rassegna di alcuni tra i più interessanti titoli software disponibili sul mercato nel settore Midi.

TIGER CUB

La peculiarità del primo programma in esame, «Tiger Cub», è la praticità d'uso dovuta alla scelta di limitare le funzioni solo a ciò che serve veramente. Non è dunque dedicato a chi deve produrre lavori di grande complessità, ma piuttosto a chi vuole usare un sequencer senza troppa fatica, e senza bisogno di imparare funzioni e comandi eccessivamente complicati.

«Tiger Cub» è, in pratica, un sequencer derivato da «**KCS**», ma con un minor numero di tracce (dodici invece di quarantotto, per la precisione); richiede almeno 1 Mb di ram per funzionare e viene fornito

su due dischetti, il secondo dei quali contiene una serie di dimostrativi e di suoni campionati pronti per l'uso.

«Tiger Cub» è in grado di effettuare la **quantizzazione** (arrotondamento delle durate e, soprattutto, dei tempi di attacco) durante ed anche dopo la registrazione, e di manipolare con notevole libertà di azione i singoli eventi Midi grazie soprattutto all'editor grafico.

Quest'ultimo, in particolare, impressiona molto positivamente: ricorda molto l'editor a barre di «Music-X», ma è più veloce e permette operazioni inusuali ed interessanti, come ad esempio una sorta di supervisione della velocità di esecuzione, gestita da quella che viene definita **Conductor Track**.

L'editor grafico visualizza le tracce, mostrando una serie di barre sullo schermo. Ogni barra corrisponde ad una nota: la sua posizione sull'asse verticale indica l'altezza, mentre la lunghezza orizzontale indica la durata.

Tramite il mouse, è semplice e veloce inserire nuove note all'interno di una traccia o modificare i parametri di quelle già presenti. I valori possono essere variati per ogni singola nota o globalmente per più note all'interno di una traccia, selezionando con il mouse quelle da editare.

Le possibilità classiche di regolazione nota per nota dell'altezza, della dinamica (**Midi Velocity**) e della durata sono separate in modo tale da controllare con il mouse solo la componente desiderata, che in alternativa si può modificare numericamente da tastiera per ottenere valori più precisi.

Si può persino lavorare in una finestra dedicata: ciò significa che, ad esempio, potremmo gestire la dinamica del brano (crescendo e decrescendo) semplicemente aprendo la finestra della Velocity (velocità di pressione di un tasto sulla tastiera Midi) e disegnando l'andamento del parametro in questione muovendoci con il mouse da sinistra a destra (seguendo cioè l'asse del tempo), oppure contemporaneamente su e giù (sull'asse della velocity). La stessa operazione si può effettuare per i parametri **Pitch Bend**, **Modulation Wheel**, ecc.

La pagina di registrazione di «Tiger Cub», un eccellente sequencer a dodici tracce della Dr. T's.

Le operazioni di «**cut & paste**» (taglia e incolla) sono estese al passaggio di note fra una traccia e l'altra. È ovviamente presente lo **zoom temporale**: premendo il tasto sinistro o destro del mouse sull'icona a forma di lente d'ingrandimento godremo rispettivamente di uno «zoom in» o di uno «zoom out».

Volendo inserire uno spartito nota per nota, ci viene in aiuto lo **Step Time Recording**, che ci consente di farlo in modo veloce suonando le note come sempre sulla tastiera midi, ma selezionando le durate tramite Amiga.

Le lunghe sequenze di note di ugual durata rappresentano i casi in cui risulta più utile questa funzione, così diffusa (per necessità, non certo per scelta) sui primi rudimentali programmi degli «antichi» computer ad 8 bit (ricordate lo Spectrum?).

Degni di nota sono anche alcuni particolari filtri derivati dal «KCS» e con i quali diventa possibile settare, scalare, invertire, incrementare o decrementare gli attributi Midi di interi segmenti di note.

«Tiger Cub» supporta il formato standard Midi per i file (una caratteristica che permette di scambiare sequenze tra computer e programmi diversi) ed i timbri IFF interni; può inoltre effettuare il **Merging** del flusso Midi in input con quello generato dalle sequenze e, come se non bastasse, viene fornito con un semplice (ma simpatico) modulo di stampa delle partiture, chiamato «**QuickScore**».

A dispetto della sua semplicità di utilizzo, soprattutto grazie all'editor grafico, e della pretesa di essere un programma per dilettanti, «Tiger Cub» è molto versatile e potente: oltre che eccellere in alcune funzioni (lo step time recording, ad esempio), è in generale davvero veloce nell'editing e dotato di funzionalità sorprendenti.

La curva nella sezione inferiore dello schermo è la Conductor Track di «Tiger Cub», che controlla la velocità di esecuzione.

BARS & PIPES

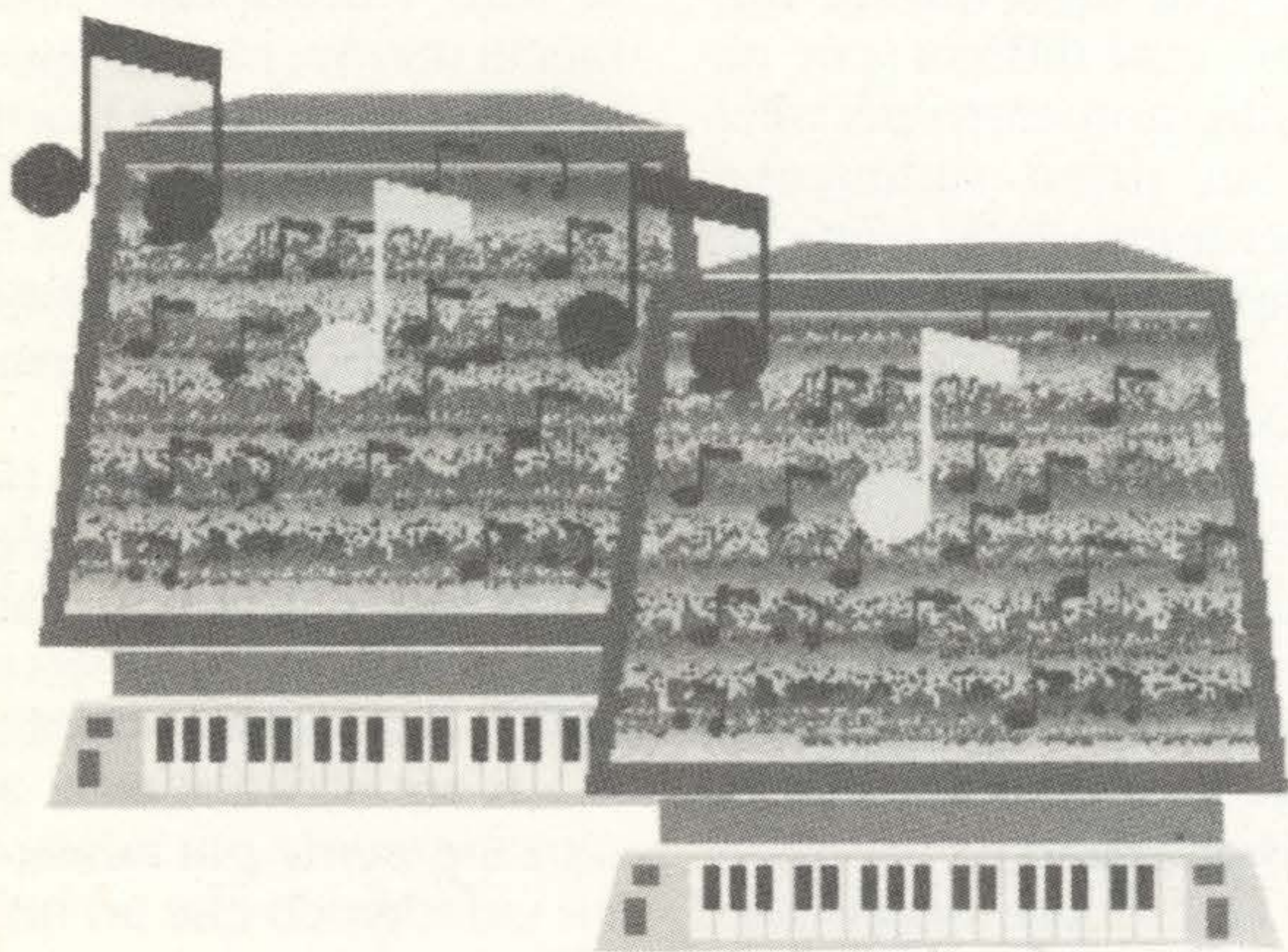
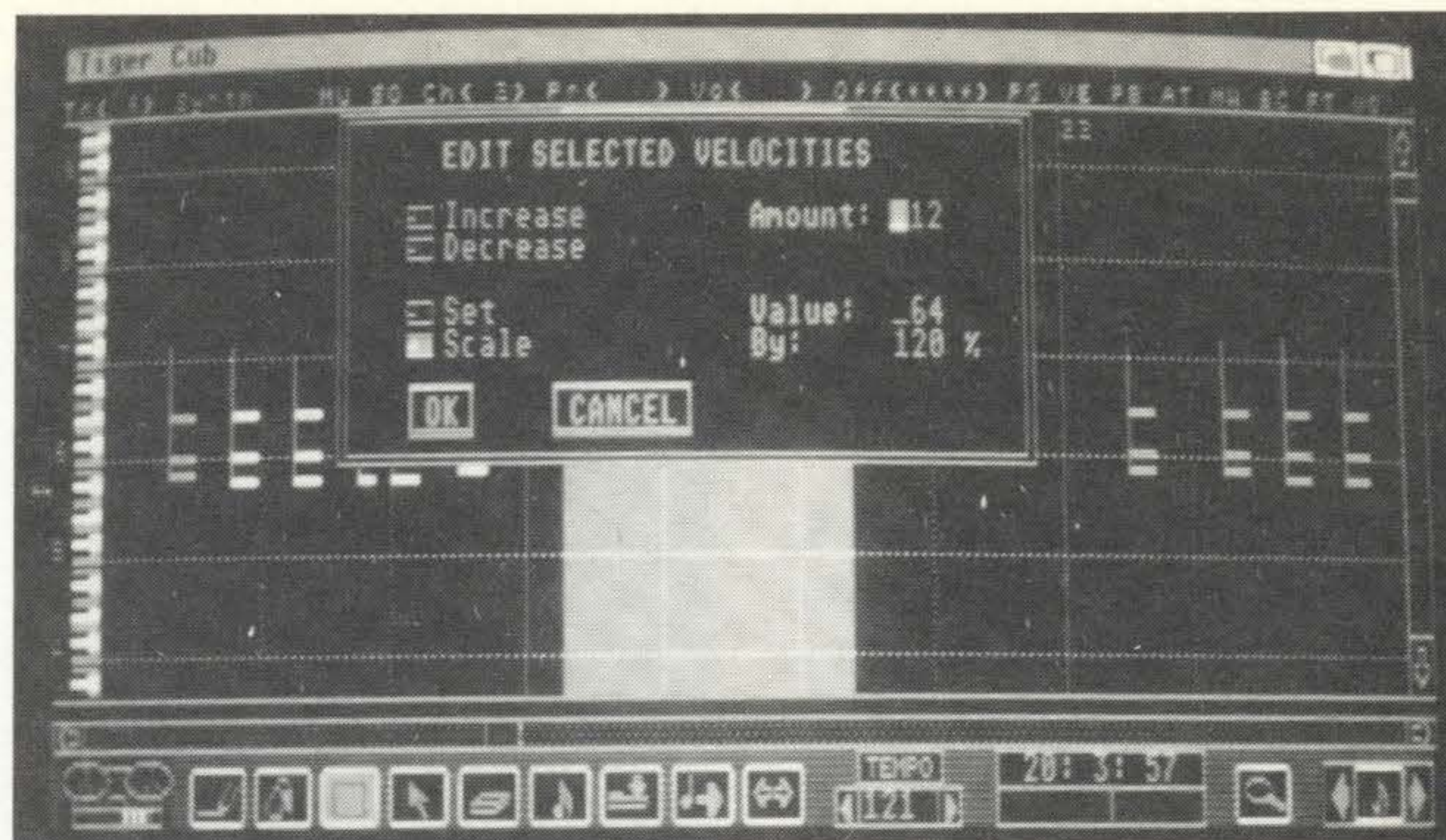
Chi è abituato ai soliti sequencer, alla loro tipica veste grafica dimessa ed alla loro tradizionale interfaccia utente, resterà quantomeno sconcertato dopo aver caricato per la prima volta «**Bars & Pipes**», il sequencer della Blue Ribbon Bakery insolito e curioso fin dal nome.

Icone dall'aspetto bizzarro, pulsanti e gadget dalle forme futuriste, rubinetti e tubature al posto dei soliti, confortanti caratteri Ascii ed altri elementi apparentemente più adatti ad un videogioco che ad un'utility musicale rappresentano il paesaggio che vi dovrete abituare ad avere sotto gli occhi utilizzando questo programma.

In ogni caso, «Bars & Pipes» possiede caratteristiche di indubbio interesse: si tratta sostanzialmente di un programma che funge da sequencer, da editor e da processore di eventi Midi, capace soprattutto di elaborazioni in tempo reale.

Le combinazioni possibili di queste elaborazioni risultano essere potenzialmente infinite, vista la possibilità di collegare tra loro (grazie alla metafora dei tubi) filtri quali il quantizzatore, lo splitter, il traspositore ed altri ancora. Provate a seguirci in un'ipotetica sessione di lavoro e forse capirete meglio il meccanismo su cui è basato «Bars & Pipes».

UNA LIBRERIA PER PROGRAMMARE

Se siete intenzionati a scrivere software che utilizzi l'interfaccia Midi, vi farà piacere sapere che esiste una libreria di funzioni che rende molto più semplice il compito di pilotare tramite software le apparecchiature Midi collegate all'interfaccia.
Questa libreria si chiama «Midi.Library» ed è di pubblico dominio; la sua versione più aggiornata è infatti reperibile sul disco 227 della libreria di programmi PD di Fred Fish.
Su quello stesso disco sono inoltre presenti alcuni programmi dimostrativi in AmigaBasic ed in C, oltre che svariati comandi richiamabili da CLI ed una serie di include e di testi di documentazione necessari per linkare le funzioni della libreria ai vostri programmi scritti in C (Lattice o Aztec).
La libreria, fra le altre cose, supporta il multitasking, ovvero

l'utilizzo contemporaneo dell'interfaccia Midi da parte di più processi, e si avvale anche delle funzioni della libreria «Arp, library», compresa nel disco.
Diamo ora uno sguardo alle utility richiamabili da CLI: non poteva mancare un comando per inviare alla porta Midi dei dati esadecimali, ed infatti questa è la funzione svolta dal comando «Ht»; il comando «Mm» invece assolve al compito inverso, interpretando e visualizzando i dati in arrivo.
Sono disponibili inoltre i comandi «Dmf» e «Playmf» che rispettivamente servono a visualizzare ed a suonare un file Midi; si tratta comunque di programmi limitati al solo supporto del formato zero (una traccia sola) e del «metrical timing». Seguono poi «Rsx» e «Tsx», che permettono l'accesso al «System Exclusive», e «Ychk», che verifica la correttezza dei file ricevuti da hardware Midi di produzione Yamaha. A titolo informativo, quest'ultima funzione non ci è parsa funzionare correttamente con il campionatore Yamaha TX16-W.
Su di un altro Fish Disk, per la precisione il numero 159, si trovano ancora altri programmi facenti uso della «Midi.library». Citiamo in particolare un generatore di eco Midi-programmabile ed un gestore di «System Exclusive» in grado di riconoscere gli errori di sintetizzatori Midi come gli Yamaha DX7 ed FB-01, o l'OberHeim Matrix6.

Questo requester di «Tiger Cub» consente di variare le impostazioni relative alla «Midi Velocity» del brano.

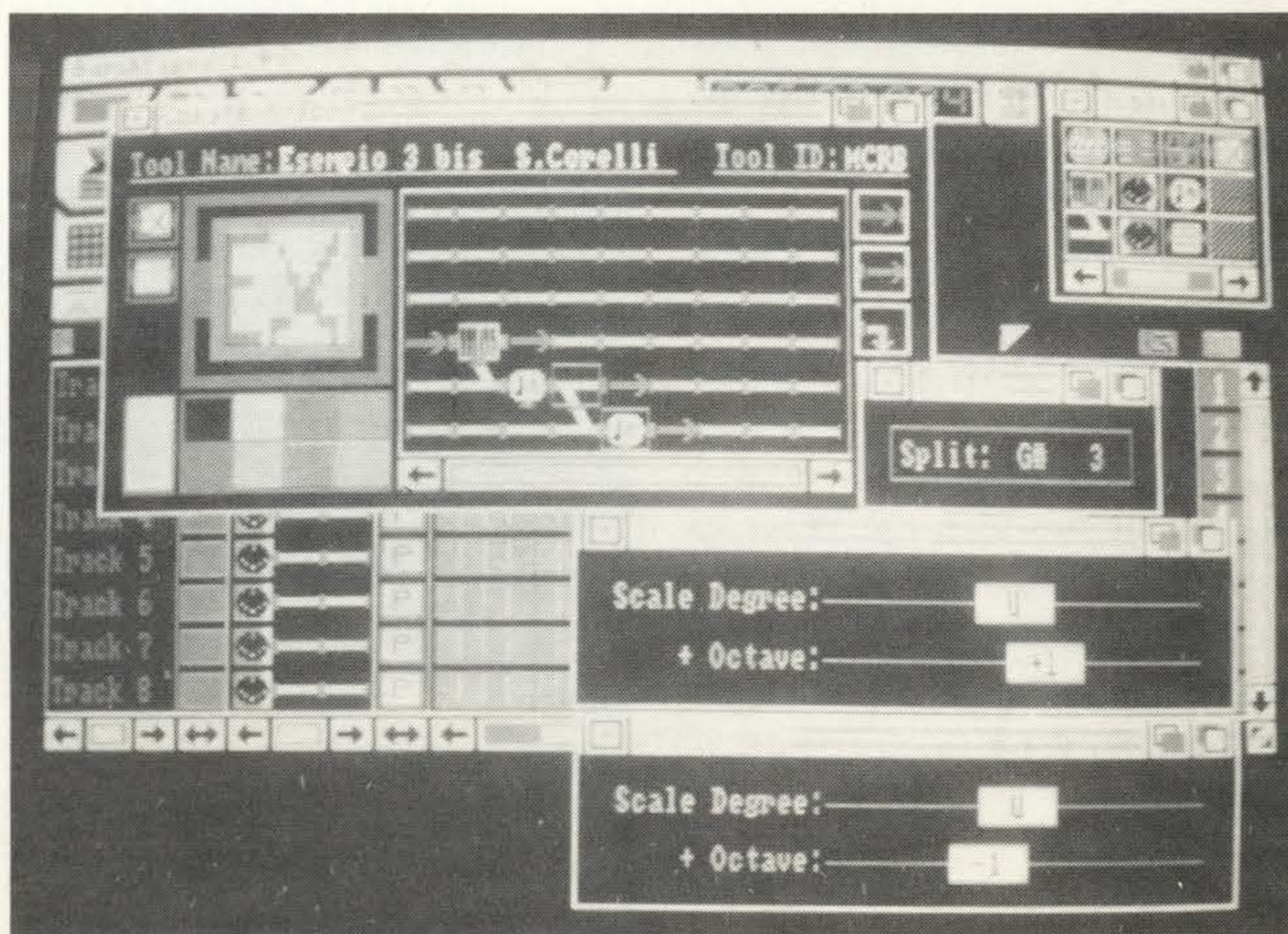
Immaginiamo di voler raddoppiare all'ottava la quinta più bassa della nostra tastiera midi. «Bars & Pipes» ci permette di farlo programmando in pochi minuti un nostro filtro personalizzato con relativa icona, che potremo poi salvare su disco riutilizzandolo in seguito, come se fosse un elemento standard del programma: piazzandolo cioè a sinistra di una traccia (influenzando irreversibilmente la registrazione), o alla sua destra (influenzandone solo il riascolto); ogni filtro potrà inoltre diventare addirittura un elemento di un nuovo macro-filtro.

In figura A potete vedere la rappresentazione a «tubi» del nostro piccolo esempio.

Un'avvertenza: quando collegate i tubi in verticale spesso non basta accostare sullo schermo i condotti, ma occorre utilizzare espressamente il comando **Connect** del **menu Pipetools** dopo aver selezionato con il mouse la connessione posta più in alto, per poi clickare su quella posta più in basso.

Ovviamente i «pipetools» (cioè i filtri) possono richiedere uno o più argomenti: per inserirli è sufficiente clickare due volte sulla relativa icona (dopo averla presa dal ToolBox e piazzata dove necessario). Inoltre, visto che alcuni filtri sono costituiti già da filtri più semplici, potrà comparire, in luogo della window di richiesta degli argomenti, lo schema di costruzione del filtro stesso, in cui potrete andare a modificare i valori dei suoi elementi costituenti (i quali potrebbero anch'essi non essere basici e far comparire un'altra window che ne rappresenta la struttura, e così via).

Si capisce dunque la difficoltà nel modificare il funzionamento di un filtro

Figura A: il nostro esperimento con «Bars & Pipes»: la creazione di un «tool» personalizzato costituito da un filtro.

complesso.

Nel nostro esempio comunque gli argomenti dei filtri sono leggibili sulle relative window in figura A.

IL MULTISPLIT

Decidiamo ora di simulare sul nostro sintetizzatore un *multisplit* (vogliamo cioè separare virtualmente la tastiera in un certo numero di sezioni capaci, per esempio, di suonare timbri diversi) ed ipotizziamo che il nostro synth sia in grado di ricevere sui primi quattro canali Midi quattro parti monofoniche ciascuna timbricamente indipendente dall'altra (un esempio classico: il synth Casio CZ-1000).

Dovremo quindi fare in modo che l'output del sintetizzatore arrivi ad Amiga e che «Bars & Pipes» ridistribuisca le note sui canali Midi 1, 2, 3 e 4 secondo le seguenti modalità: le note sotto il punto di split più basso dovranno andare al canale 1; quello comprese tra il primo ed il secondo punto di split dovranno andare invece al canale 2; quelle tra il secondo ed il terzo split dovranno essere redirezionate al canale 3 e, per finire, le note al di sopra del punto più alto di split dovranno giungere al canale 4.

Questo secondo esempio crea grossi problemi visto che non si possono manipolare i canali Midi con alcun filtro. La soluzione è fornita dal meccanismo di connessione di «Bars & Pipes», che si applica anche alle varie tracce; potremo quindi connettere input ed output (filtrati o meno) di tracce diverse e, visto che ogni traccia indirizza un preciso canale midi a scelta, ci basterà appunto connetterne quattro utilizzando i tools **Note Filters**, **Branch** e **Merge** per raggiungere il nostro scopo.

Immaginate quali strani tool potrete creare manipolando e connettendo tra loro tool quali **Delay**, **Echo**, **Articulator**, **Quantize**, o il bizzarro **Invert Pitch**.

Probabilmente a qualcuno sarà venuto in mente il «**Pro Midi Studio**» della SoundScape, che crediamo abbia ispirato i programmatori di questo «Bars & Pipes»: anch'esso infatti mostra i collegamenti fra porte Midi, sequencer e generatore sonoro interno, tracciando linee che uniscono le icone sul video.

Di positivo rimane da dire che il programma gestisce il **protocollo Smpte** ed il **formato Midi** per i file, mentre di negativo c'è purtroppo da rilevare che l'editor grafico di eventi si è rivelato estremamente lento e confusionario (è quasi meglio quello, decisamente primitivo, del vecchissimo programma «**The Music Studio**» dell'Activision); inoltre, non si possono usare i suoni IFF di Amiga, tanto comodi per i piccoli lavori sulle sequenze preregistrate.

L'editor dei tool di «Bars & Pipes», pur non essendo del tutto esente da bug (almeno nella versione da noi provata), rappresenta senza dubbio un'ottima idea e rende inoltre il programma facilmente espandibile. Tuttavia, globalmente rimane l'impressione di avere a che fare con un software ancora non completamente riuscito, dall'aspetto molto «ludico» ma poco professionale: la mancanza di una rappresentazione delle note in formato musicale standard oltre che a barre, ad esempio, rende a volte difficile distinguere a prima vista le durate delle diverse note, specie quando sono molto brevi.

La Blue Ribbon Bakery ha comunque annunciato l'imminente uscita di numerosi *upgrade* per «Bars & Pipes», tra i quali vari tool per la creazione di arpeggi o per il mix-down: non resta quindi che attendere e sperare in un'eventuale nuova versione potenziata.

THE COPYIST

Se avete l'esigenza di stampare partiture musicali e possedete un Amiga ed una stampante Epson compatibile, allora non dovete farvi sfuggire «**The Copyist Professional 1.6**», sempre prodotto dalla **Dr. T's**, un vero e proprio programma di **Desktop Publishing musicale**.

Colpisce in particolare, di questo software l'elevata qualità grafica ottenibile appunto con le tradizionalmente economiche stampanti Epson-compatibili che, come noto, si avvalgono di soli nove aghi: grazie alla tecnica di stampa grafica «interlacciata», usata anche da «**Page Stream**» e da altri analoghi programmi di DTP, si raggiungono risoluzioni esteticamente quasi comparabili a quelle ottenibili con stampanti laser. Una dimostrazione di quanto vi diciamo la potete avere osservando la stampa riportata in figura B, ottenuta con una Epson LX800.

Figura B: una bella stampa di uno spartito ottenuta con «The Copyist» ed una normale stampante ad aghi Epson LX800.

Per poter raggiungere questi risultati qualitativi, «The Copyist» non usa i driver di stampa forniti con il WorkBench 1.3: la stampa avviene infatti esternamente dal programma principale, tramite una serie di moduli personalizzati, da selezionare in relazione al tipo di stampante usata.

L'utilizzo di «The Copyist» non è tra i più intuitivi, causa la grande quantità di comandi disponibili. Anche se molti di essi sono duplicati sui menu a tendina, i comandi devono essere preferibilmente inseriti con la tastiera: cosa che

L'aspetto di «Bars & Pipes» è sicuramente inconsueto rispetto ai tradizionali sequencer Midi.

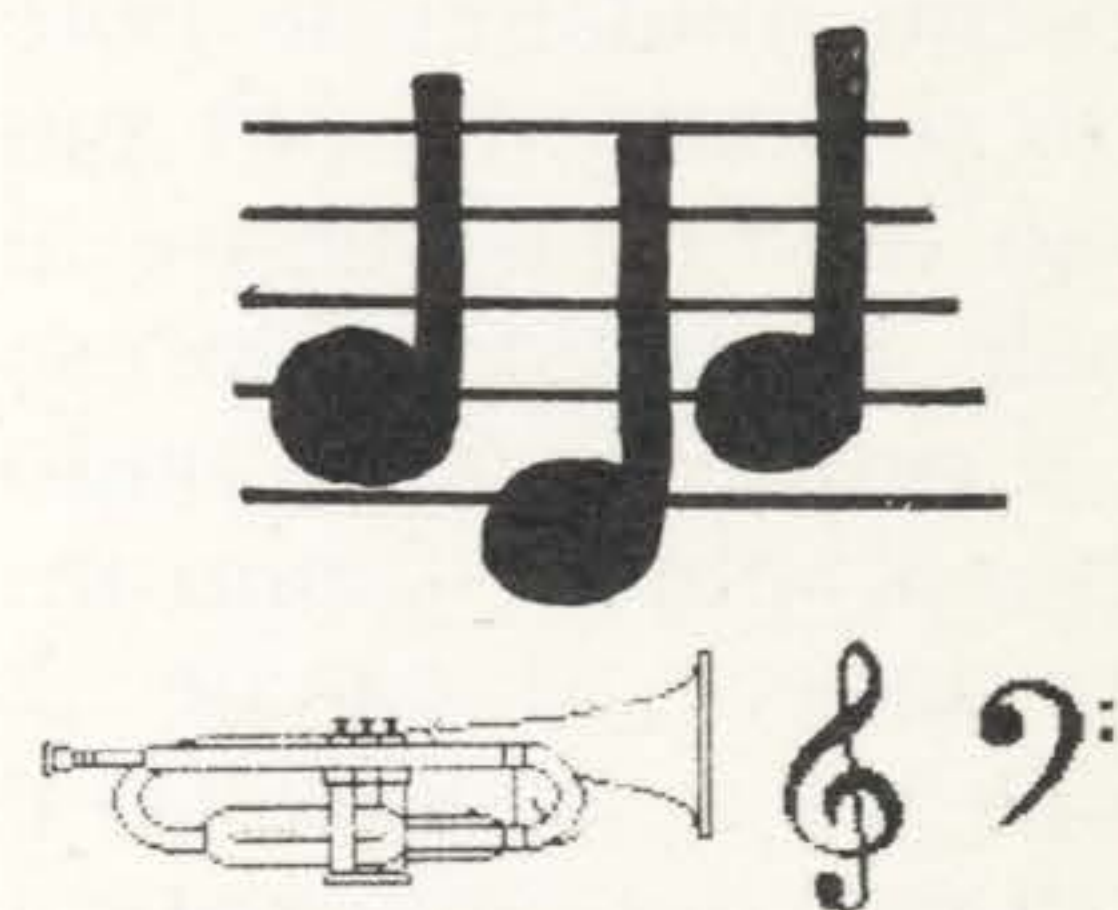
consente, a parte l'iniziale fatica di apprendimento mnemonico, una maggiore velocità e precisione nell'inserimento delle note.

Un piccolo esempio per comprendere meglio la filosofia di interazione sulla quale si basa «The Copyist»: apriamo un documento con il **comando New** del **menu Project**, creiamo un pentagramma premendo la lettera «**S**» (maiuscola) e, con le due lettere «**ct**» (minuscole), facciamo comparire la chiave di Sol. Proviamo ora a muoverci sul rigo tramite il mouse o i tasti cursore, ed inseriamo le note al loro posto con il tasto «**w**» per i pallini vuoti e «**b**» per quelli pieni, facendo seguire il tipo di asta, verticale o orizzontale, rappresentante valori fino al sessantaquattresimo con la pressione dei tasti appropriati.

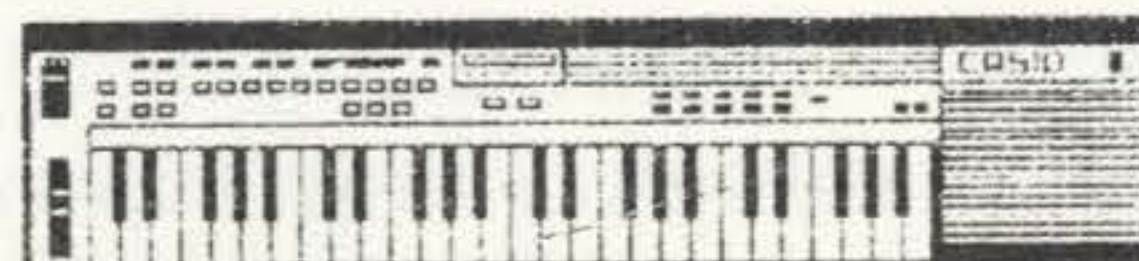
Il **tasto Help** (finalmente un programma che lo sfrutta !) fornisce un esauriente riepilogo di comandi e tasti da usare. Sono in ogni caso possibili numerose operazioni, a partire dalla notazione per gli accordi della chitarra (o per i testi delle canzoni) fino ad arrivare ai simboli musicali di crescendo e diminuendo.

Risulta particolarmente utile l'interessante capacità di importare file in formato Midi o nel formato .ALL usato dagli altri programmi della Dr. T's, come il già citato «KCS». Potete quindi realizzare una composizione musicale con il vostro sequencer preferito e farla stampare a «The Copyist», eventualmente suddividendola tramite il **modulo Parts** e stampando separatamente le partiture per ogni singolo strumento. Viceversa, tramite il **modulo Synth** è possibile convertire la partitura disegnata con l'editor di «The Copyist» in un file in formato Midi standard, da caricare ed in seguito far eseguire al sequencer.

IL FONT EDITOR

Sul disco sono presenti file dimostrativi e vari moduli tra cui un editor di simboli grafici per la stampante e suddivisori di partiture in spartiti per i singoli strumenti. Il **Font Editor** è utile per personalizzare all'occorrenza i simboli

musicali usati nelle partiture, ma è limitato ad un massimo di dieci caratteri customizzati.

Esistono altre due versioni di «The Copyist»: la prima è chiamata «**The Copyist Apprentice**» ed è, in pratica, una versione ridotta e semplificata rispetto a quella professionale, che consente un output massimo di cinque pagine alla volta (in contrapposizione alle 50 della versione «Pro»); la seconda è chiamata «**The Copyist DTP**» e si distingue per la presenza del **supporto PostScript** per stampanti laser, del font strutturato di caratteri musicali «**Sonata**» della Adobe Systems, e della possibilità di stampare partiture lunghe fino ad un massimo di 100 pagine.

MIDI GLOSSARIO

MIDI: Abbreviazione di «Musical Instrument Digital Interface», è uno standard per la comunicazione di dati tra uno strumento musicale ed il computer. Normalmente le interfacce Midi su Amiga vengono collegate tramite la porta seriale, ed i dati vengono trasmessi alla velocità di 31250 bit al secondo. AmigaByte ha presentato un progetto per autocostruire un'interfaccia Midi sul fascicolo 2.

SYSTEM EXCLUSIVE: Il System Exclusive è una parte del protocollo di comunicazione Midi usata per la trasmissione di dati personalizzati, relativi a particolari strumenti musicali ed al loro software dedicato. Si tratta in pratica di messaggi che viaggiano su di un canale riservato, tramite il quale hardware e software possono comunicare via Midi anche in presenza di altre apparecchiature Midi incompatibili.

SMPTE: Abbreviazione di «Society of Motion Picture and Television Engineers», è un tipo di notazione standard sviluppata dal comitato omonimo per la sincronizzazione di musica ed immagini. Viene usata solitamente nella realizzazione di colonne sonore. Il tempo in formato SMPTE viene misurato in fotogrammi, oltre che in minuti e secondi.

TRACCE DIFETTOSE

Ho acquistato recentemente un hard disk A590 e l'ho aperto, seguendo le istruzioni riportate nel manuale, per inserire 16 chip di ram ed espandere di due megabyte la memoria del mio Amiga. Durante questa operazione ho notato che sull'hard disk Epson è presente un'etichetta con la scritta «Defect List», seguita da una sfilza di numeri di tracce e testine. Presumendo che essa indicasse le tracce difettose del disco, mi sono rivolto al rivenditore per la sostituzione, ma lui non ha voluto saperne affermando che la cosa è perfettamente normale.

Mario Cerrone - Bergamo

Praticamente nessun disco rigido viene prodotto senza errori od imperfezioni: la superficie del supporto magnetico quasi sempre presenta tracce occasionalmente difettose, che vengono rilevate durante il controllo di qualità eseguito in fabbrica e segnalate sull'etichetta fissata al drive. Questo è perfettamente normale: la segnalazione viene messa per consentire all'utente di contrassegnare manualmente le tracce danneggiate nella tabella di allocazione dei blocchi del disco, in modo che esse non vengano usate per memorizzarvi dei dati. Questa operazione viene comunque normalmente svolta in automatico dalla funzione di verifica della superficie del disco compresa nel programma «HdToolbox» fornito con il drive.

La presenza di «bad blocks» rientra perciò nella norma e non dovrebbe causare problemi nell'utilizzo dell'hard disk.

SOFTWARE INTROVABILE

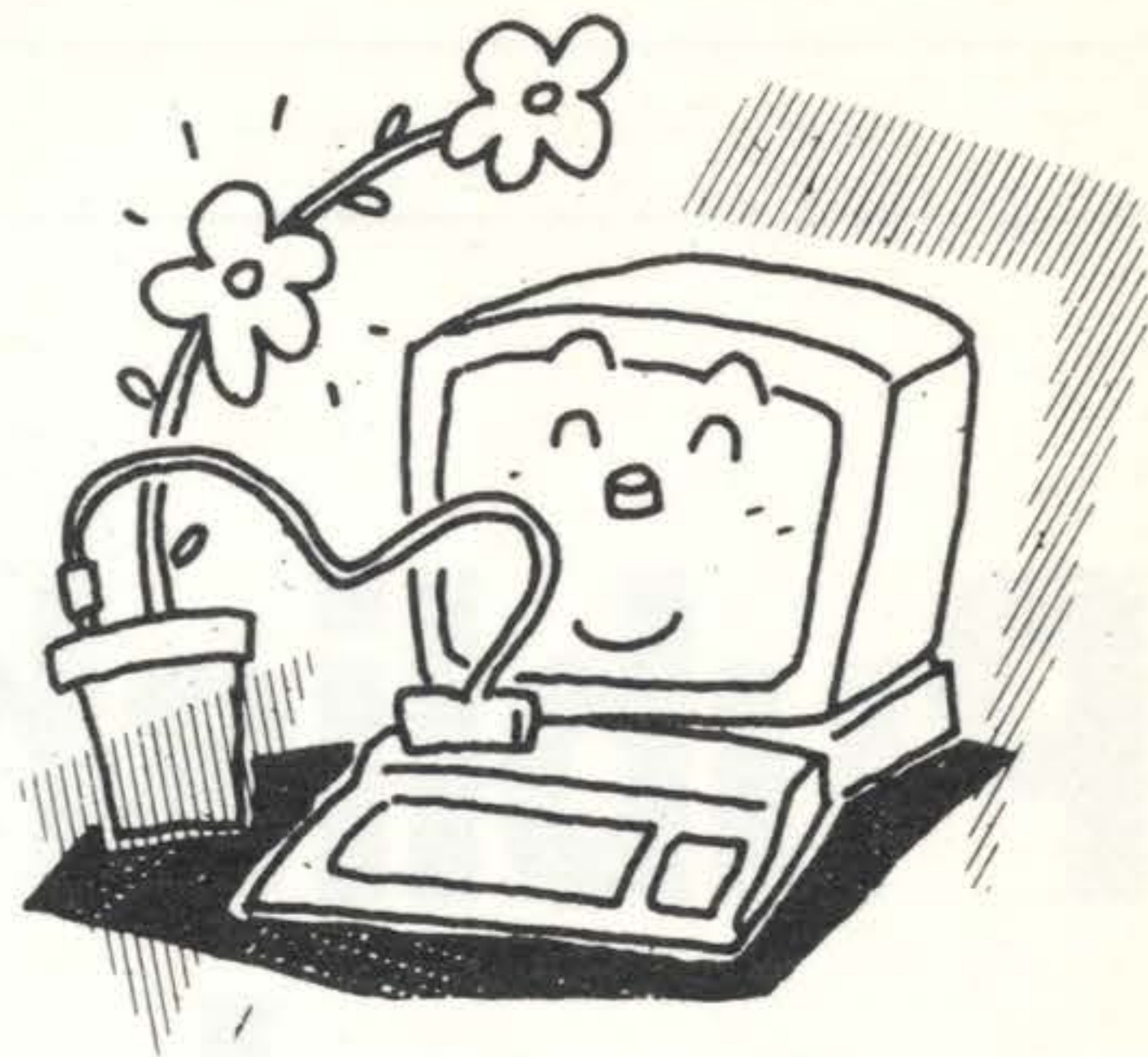
Sono un appassionato di avventure testuali, in modo particolare quelle della Infocom. So che la Infocom ha convertito la trilogia di Zork anche per Amiga: io ho trovato solo Zork III ma non Zork I o Zork II. Mi potete dare qualche indicazione?

Sante Camporesi - Forlì

Anche il software, esattamente come accade nel campo dell'editoria, non resta commercializzato in eterno: i programmi più vecchi vengono gradualmente messi fuori catalogo e non più distribuiti dalle software house.

La Infocom, inoltre, non esiste più (è stata assorbita dalla Activision ed ha cessato di produrre adventure testuali): la maggior parte dei suoi celebri giochi è apparsa anche su Amiga intorno al 1986, ma è ora quasi scomparsa dal mercato. Solo di recente l'inglese Mastertronic ha rimesso in circolazione, in una speciale serie economica, cinque titoli Infocom per Amiga al prezzo di 9,99 sterline.

I giochi sono «Zork I», «The Hitchhiker's Guide to the Galaxy», «Leather Goddesses of Phobos», «Planetfall» e «Wishbringer».

Una ditta inglese specializzata nella vendita di adventure e giochi di simulazione ha inoltre in catalogo gran parte dei giochi Infocom a prezzi molto convenienti. Per maggiori informazioni, puoi rivolgerti presso Special Reserve, P.O. Box 847, Harlow, CM21 9PH, Gran Bretagna.

UN BUG DEL WORKBENCH

Avendo appena acquistato un Amiga 500, ho seguito le istruzioni riportate nel manuale ed ho tentato di effettuare una copia di sicurezza del dischetto WorkBench incluso nella confezione: ho clickato una volta sull'icona del disco «WorkBench1.3» ed ho selezionato l'opzione Duplicate dal menu in alto a sinistra. Nonostante tutti i miei ripetuti tentativi, non sono riuscito a venire a capo di niente perché il dischetto non si copia. Dove sbaglio?

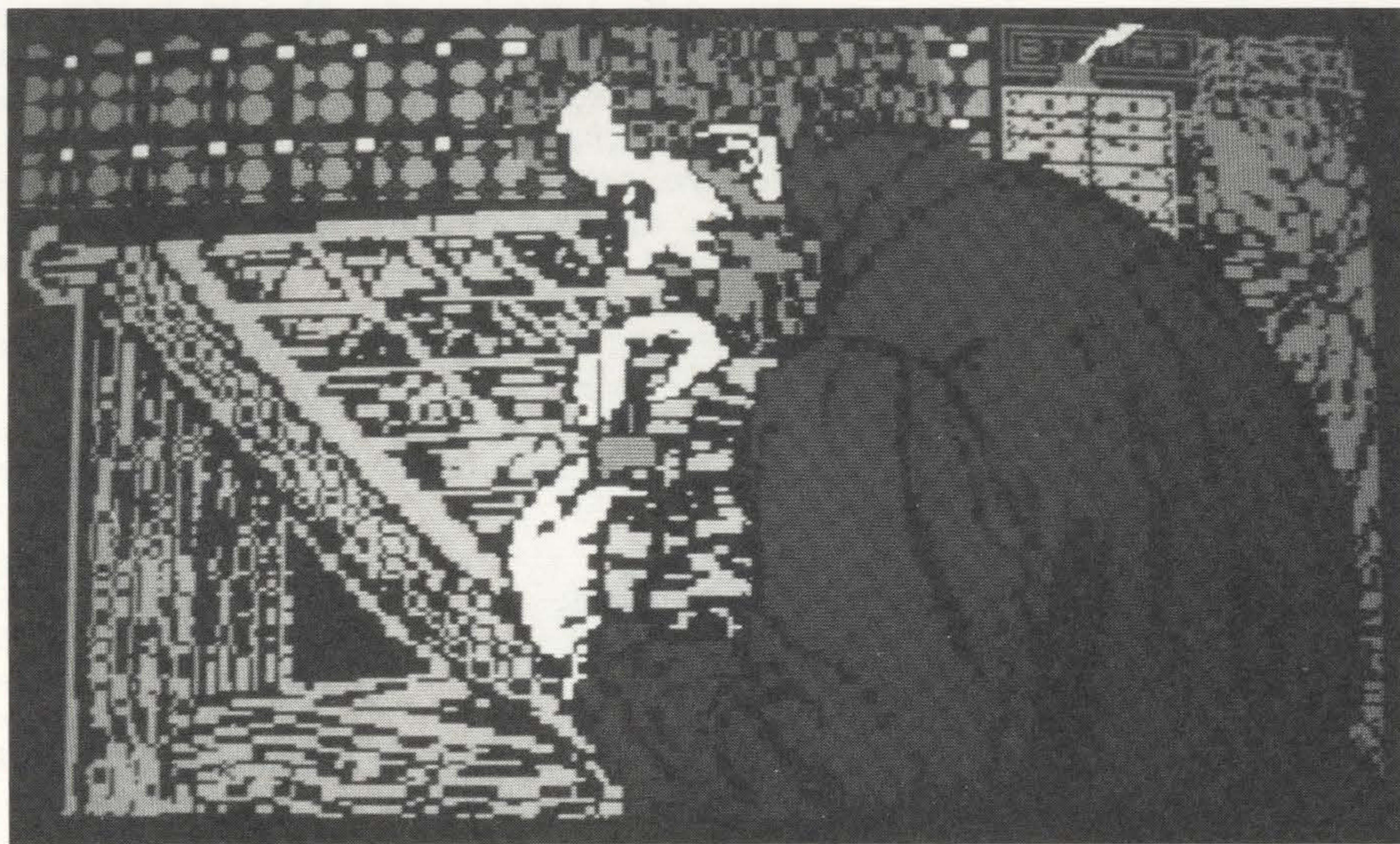
Luciano Bevilacqua - Sarzana

Apparentemente stai utilizzando la versione 1.3.2 del WorkBench, che sembra contenere un bug legato proprio alla duplicazione dei dischetti. Per copiare un disco, la cosa migliore è ricorrere al comando «DiskCopy», utilizzando il Cli. È sufficiente clickare sull'icona «Shell» o sulla «Cli», contenuta solitamente nel cassetto «System», e digitare questo comando:
Diskcopy Df0: to Df0:

Il computer richiederà inizialmente l'inserimento del disco «WorkBench» originale e poi quello del disco destinazione (che non deve necessariamente essere stato formattato in precedenza). L'operazione andrà ripetuta più volte, ed al termine il disco destinazione conterrà una copia perfettamente funzionante del «WorkBench1.3».

Publishing Partner Master contro Saxon Publisher

A confronto due sofisticati pacchetti di DTP per Amiga. Uno scontro al vertice che sancisce una duplice vittoria

di FRANCESCO OLDANI

Nell'ambito del Desktop Publishing i programmi per Amiga si contano comodamente sulle dita di una sola mano, ed il più noto è «**Professional page**» della Gold Disk, recentemente giunto alla versione 2.0. Questo eccellente programma ha detenuto un'incontrastata leadership nel settore perchè, fino a circa un anno e mezzo fà, era l'unico pacchetto che permettesse un impiego professionale di Amiga nel settore del DTP. Nel 1988 è giunto sul mercato «**PageStream**», un programma in grado di contrastare, e in alcuni casi superare, «Professional Page» sul piano della potenza e della versatilità, anche se a prezzo di un'affidabilità inferiore (dei due software citati Amiga Byte si è occupata nei fascicoli 15 e 24.

A questi due programmi se ne è aggiunto oggi un terzo, di confrontabile levatura: «**Saxon Publisher**», che promette tantissimo, tanto da presentarsi nelle inserzioni pubblicitarie come il nuovo punto di riferimento. Visto che quasi in contemporanea è giunta sul mercato la versione potenziata di «PageStream» ribattezzata dalla SoftLogik «**Publisher Partner Master**», può essere interessante un confronto tra questi due ultimi pacchetti.

I CONTENDENTI

«Publisher Partner Master» (che d'ora in avanti chiameremo «PPM») è, come già detto, la versione europea di «PageStream 2.0». Le caratteristiche di punta di questo programma (messo a disposizione da DigiMail Srl, Via Coronelli 10, 20146 Milano) sono l'elaborazione di più documenti contemporaneamente, una gestione del testo molto potente, l'utilizzo dei font strutturati **AGFA Compugraphics**, la possibilità di creare disegni strutturati comprendenti curve di Bezier, una precisione che arriva a 1/100 di grado nelle rotazioni degli oggetti, ed infine la possibilità di gestire documenti lunghi fino a 1000 pagine (!).

«Saxon Publisher» è il primo programma della Saxon Industries, ed è un prodotto molto innovativo rispetto ai pacchetti concorrenti. Il testo e le imma-

gini possono essere contenuti in un box di qualsiasi forma, e si possono inoltre definire degli sfondi attraverso un sofisticato sistema di retinatura chiamato **Texture**.

Ma, oltre che queste caratteristiche uniche, quello che rende «Saxon» un programma originale è il suo modo di operare. La filosofia di progetto è basata sull'utilizzo esteso dei **Tag**, ovvero di componenti predefiniti e richiamabili con la sola pressione di un tasto, o con il mouse.

«Saxon» permette di definire per ogni oggetto sulla pagina un Texture, cioè uno sfondo grafico, anche molto complesso.

INSTALLAZIONE

«PPM» è memorizzato su quattro dischi: il primo contiene il programma, il secondo i driver di stampa, e gli ultimi due i dischi font. «Saxon» è memorizzato su due dischi. Occorre osservare che per utilizzare in maniera produttiva i due programmi occorre avere un Amiga dotato di hard disk e di almeno 2 Mbyte di ram, nonostante le case produttrici dichiarino che i loro programmi lavorano anche con un solo Mbyte; inoltre, «Saxon» è in grado di stampare unicamente su stampanti laser PostScript. Si capisce che i due programmi sono riservati a professionisti che hanno configurazioni hardware ben espanse (e costose). «PPM» invece stampa molto bene anche con comuni stampanti ad aghi, o comunque non Postscript.

Di default entrambi i programmi lavorano su di uno schermo custom interlacciato con 16 colori; con «PPM» si può optare anche per un numero di colori inferiore a 16 (4 o 8). Molto vorace di memoria «Saxon», che occupa alla partenza ben 2 Mb di fast ram e 350 Kb di chip ram su di un sistema con 3.5 Mb totali. Sullo stesso Amiga, «PPM» si accontenta di 600 Kb di fast ram e di 250 Kb di chip ram.

Con «PPM» la prima cosa da fare è fissare tutti i path per i documenti, i testi, le immagini grafiche, i driver ed i font. Da questo primo impatto ci si rende conto della complessità del programma e dell'elevato numero di file esterni che occorrono al pacchetto per funzionare correttamente. I font in particolare sono gestiti in maniera non convenzionale. Attraverso un'opzione, detta **Manager**, è possibile selezionare le directory in cui sono memorizzati i font che il programma utilizzerà. In questo modo si effettua una suddivisione razionale dei font in più directory. Altra procedura da eseguire è la selezione del driver della stampante, in quanto «PPM» non utilizza i driver del WorkBench. Se tutti gli assegnamenti descritti sono stati effettuati correttamente, il programma è pronto a funzionare.

Nell'ultimo **menu Global** sono presenti anche altre preferenze tra cui la definizione delle «**Macros**» (sequenze di comandi che, associati ai tasti funzione, permettono di eseguire intere operazioni con la pressione di un tasto), e dell'unità di misura selezionabile tra i nove tipi disponibili. Tutti questi parametri, una volta selezionati, andranno memorizzati su disco una volta per tutte.

Con «Saxon» le cose sono molto più semplici, ma non in senso positivo: infatti i path vanno indicati di volta in volta attraverso le finestre di dialogo quando si caricano i vari file, operazione che rallenta il lavoro. Gestendo solo stampanti PostScript, la scelta dei driver si riduce tra il driver di default e quello denominato «**lino1331pi**» che si riferisce ad una macchina tipografica.

Se la sola gestione delle stampanti PostScript può essere considerata una scelta di progetto, una grossa nota di demerito va al fatto che «Saxon» utilizza come unità di misura esclusivamente i pollici, il che mette in difficoltà l'utente abituato al sistema metrico decimale. Altra cosa grave è la gestione interna della tastiera che non consente l'inserimento di vocali accentate in modo semplice: è infatti utilizzabile solo la tastiera americana.

Dal punto di vista della versatilità «PPM» è nettamente superiore, consentendo all'utente di preparare l'ambiente di lavoro secondo le proprie esigenze.

La definizione della pagina con «PPM» avviene tramite la finestra in figura. Il numero delle pagine pre-definite è molto elevato.

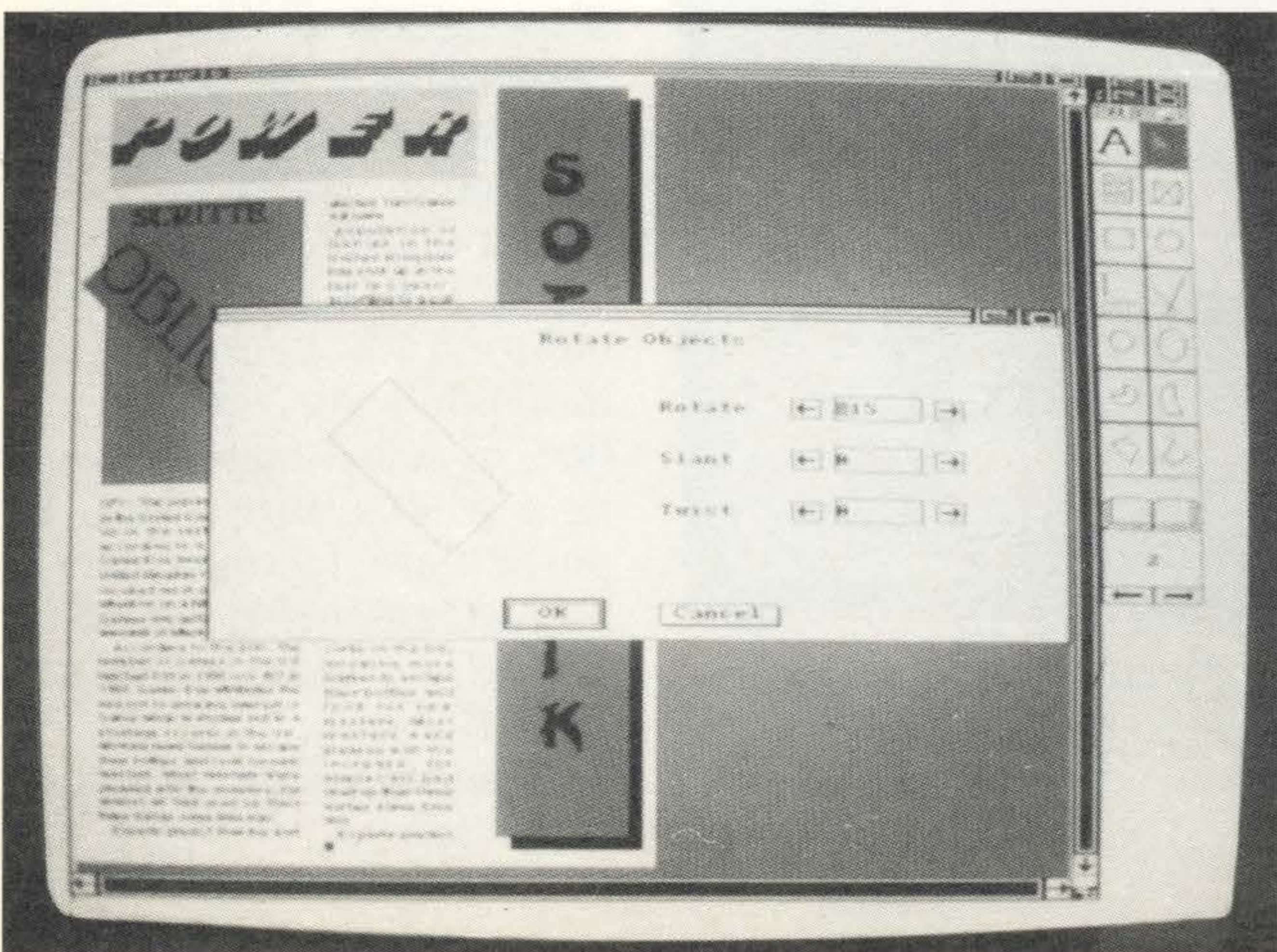
GLI STRUMENTI PRINCIPALI

Entrambi i programmi dispongono di due barre contenenti le diverse icone relative ai tool di uso più frequente. Nella barra di comando di «PPM» sono presenti le icone che permettono di passare dalla modalità testo a quella grafica, di selezionare le pagine e di costruire i box. In particolare, l'icona relativa al tracciamento delle curve aziona la routine per le curve di Bezier.

Manca lo slider per spo-

Un testo diviso in colonne con «PPM». È possibile stabilire se il testo presente in una colonna debba o meno proseguire in un'altra.

starsi sulla pagina quando questa è ingrandita, in quanto ogni pagina è racchiusa in una finestra dotata degli appositi slider come le normali finestre di Intuition. «PPM» consente di caricare più documenti, ognuno nella sua finestra: questo è un altro punto di forza del programma che non ha riscontro nei concorrenti. La barra di comando di «Saxon» invece presenta un numero maggiore di icone ed è organizzata in maniera diversa. Le modalità di lavoro sono quattro: testo, grafica, blocchi di testo, creazione e manipolazione dei box. Ogni qualvolta si seleziona una modalità diversa, le icone posizionate nella parte inferiore cambiano, rendendo disponibili gli strumenti relativi alla modalità selezionata.

Quando si costruisce un box, oppure quando si effettua l'impaginazione di un blocco di testo, o si disegna un elemento grafico, è possibile memorizzarne le caratteristiche. L'operazione è effettuabile semplicemente clickando sull'**icona New** dopo aver attivato l'elemento che interessa: questa richiamerà una finestra nella quale si inserirà il nome del tag così definito. Il nome assegnato verrà mostrato nella tabellina posta sulla barra di comando sotto il gadget per lo spostamento della pagina. Quando si vorrà ricostruire un qualsiasi oggetto (grafico o testuale) con le stesse caratteristiche di quello memorizzato, sarà sufficiente utilizzare il tag memorizzato.

Con due sole pressioni del tasto sinistro del mouse, la prima sull'oggetto e la seconda sulla tabellina in cui è annotato il tag, si otterrà il risultato desiderato. Il tutto avviene in ogni modalità operativa: addirittura è possibile assegnare ad ogni box un tasto funzione, in modo da richiamarlo con la sola pressione del tasto.

Per quanto riguarda il testo, tutte le operazioni possibili con «Saxon» possono essere effettuate anche con «PPM»; quest'ultimo inoltre offre un numero maggiore di opzioni, utili per impieghi avanzati. Sembra non si finisca mai di esplorare le opportunità fornite da «PPM»; da quasi ogni finestra è possibile aprirne un'altra contenente altri comandi.

È poi notevole quanti attributi possono essere assegnati al testo, e che molti di questi siano personalizzabili attraverso gli editor interni. In definitiva, la potenza di «PPM» e la facilità operativa con cui si lavora con «Saxon» sono davvero impressionanti.

CREAZIONE DI UN DOCUMENTO

Per creare un documento con «Saxon» si seleziona **l'opzione Add** del **menu Page**. Comparirà una finestra in cui sarà possibile fissare una qualsiasi dimensione della pagina, oppure sceglierne una di quelle predefinite (Letter, Business Card, Legal, A3 e A4). Oltre che le icone, che consentono di impostare i parametri sopra menzionati. Ed è in questa finestra che si incontra la novità che la «Saxon» ha introdotto per prima in un pacchetto di Dtp; attraverso l'**icona Define Background Texture** si accede ad un'ulteriore finestra, molto più grande e complessa della precedente.

Il texture di una pagina non è altro che lo sfondo della pagina stessa: per fare un esempio, un normale quaderno a quadretti per uso scolastico è composto da un insieme di pagine che hanno come sfondo una retinatura a quadretti con maglie più o meno larghe.

Solitamente, qualsiasi programma di Dtp permette di fissare dei pattern (figure) che, posti sulla pagina, danno l'effetto voluto.

I Texture di «PPM» sono meno sofisticati che con «Saxon»; si può scegliere tra un vasto numero di pattern con la possibilità di definirne altri.

Uno strumento potentissimo offerto da «PPM» è quello che consente di ruotare lungo tre assi qualsiasi oggetto presente sulla pagina.

Nel caso si desideri avere uno sfondo particolarmente elaborato, i normali pattern possono rivelarsi insufficienti; «Saxon» consente invece di fissare dei Texture molto complessi, che possono essere di tre tipi: trasparenti, solidi o di tipo bitmap (cioè definiti con dei pattern).

Ad ogni texture è associabile un colore, si può fissare l'effetto di ombreggiatura che eventuali oggetti avranno sulla pagina. Il concetto di texture qui brevemente spiegato per le pagine è estendibile senza differenze a testo e box. Per questi ultimi si possono inoltre fissare i colori dei contorni. Tutte queste funzioni combinate offrono l'opportunità di creare documenti così complessi ed elaborati da poter dare qualche problema in fase di stampa. Per ovviare a ciò esiste un'opzione chiamata **Trap** che, agendo in fase di separazione dei colori, limita evetuali imperfezioni di riproduzione. Effettuate le scelte desiderate, la pagina creata apparirà sullo schermo di lavoro pronta per essere elaborata.

Con «PPM» la creazione di un nuovo documento avviene attraverso l'**opzione News** del primo menu. Essa richiama una finestra dalla quale è possibile scegliere tra ben undici misure, e definire una pagina con misure a piacere. «PPM» consente di definire una pagina orientata anche per il lato maggiore (**Landscape**) e memorizzare pagine «mastro», cioè pagine pre-costruite da utilizzare tutte le volte che si deve realizzare una serie di documenti con lo stesso tipo d'impaginazione. È possibile inoltre lavorare con pagine destre e sinistre per realizzare, ad esempio, documenti con pagine simmetriche. Texture a parte, «PPM» si rivela molto più versatile di «Saxon» nella definizione delle pagine.

«Saxon» permette di effettuare vari ingrandimenti della pagina in elaborazione, anche con incrementi a scelta.

Un'altra delle caratteristiche importanti di «PPM» è la **funzione Append**, che permette di fondere più documenti separati in un unico documento.

L'IMPAGINAZIONE

Definite le pagine che compongono il documento, l'operazione successiva è quella di preparare il «**layout**» della pagina, cioè disporre i box che conterranno grafica e testo. Prima occorre però disporre tutti i box.

Con «PPM» i box di testo possono essere creati molto facilmente clickando sull'icona relativa posizionata nella barra di comando e tracciando successivamente con il mouse l'area di testo. Il metodo più potente per creare i box di testo è quello fornito dall'"**opzione Create Columns** del **menu Layout**: quest'opzione richiamerà una finestra dalla quale sarà possibile impostare le dimensioni dei box, il numero delle colonne di testo, la distanza tra queste ultime ed il linker automatico del testo. Attivando questa opzione, quando il testo contenuto in una colonna avrà esaurito lo spazio a disposizione verrà posto nella colonna adiacente.

Il testo sulla pagina può essere immesso anche direttamente senza creare nessun box. Scrivendo sulla pagina, «PPM» creerà un box dinamico che assumerà le dimensioni del testo. In questo modo titoli ed intestazioni potranno essere posti sulla pagina direttamente.

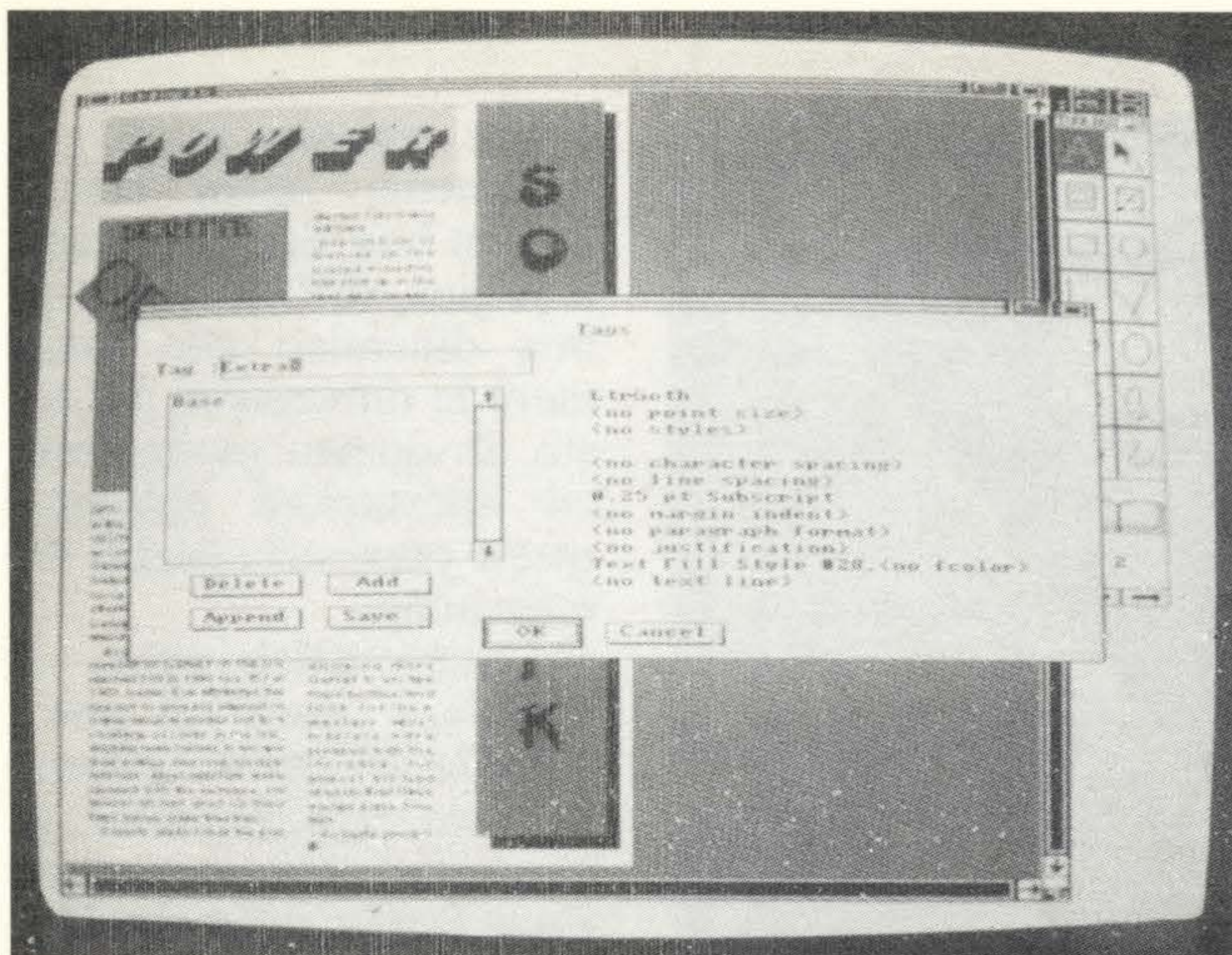
Caratteristica unica di «PPM» è quella di poter considerare ogni box come fosse un oggetto grafico. In particolare, attraverso l'**opzione Rotate** del **menu Objects** è possibile ruotare sui tre assi un qualsiasi box con tutto il suo contenuto, consentendo di porre sulla pagina scritte oblique con qualsiasi angolazione. Non è necessario porre sulla pagina i box per le immagini, in quanto questi verranno creati automaticamente quando le immagini verranno importate.

È possibile porre sulla pagina disegni vettoriali utilizzando gli strumenti interni del programma, per tracciare linee rette e curve (di Bezier), rettangoli, cerchi, etc. Sempre nel **menu Objects** sono presenti alcune opzioni tipiche dei programmi di CAD quali **Group**, **Ungroup**, **Look**, **Allign**, e così via.

Per agevolare il posizionamento dei box ed il tracciamento di motivi grafici, sono disponibili alcuni comandi comuni a tutti i pacchetti di DtP. Non comune è invece la versatilità di utilizzo: è possibile creare una griglia di riferimento sulla pagina, definendone il passo attraverso una finestra che consente di scegliere tra diverse misure pre-impostate, oltre che fissarne una con maglie di dimensioni qualsiasi.

Attraverso l'**opzione Snap to grid** si può far aderire perfettamente ogni oggetto alla griglia; si possono creare anche delle guide per un più veloce lavoro di composizione.

Nel **menù View** sono contenuti i comandi per l'ingrandimento della pagi-

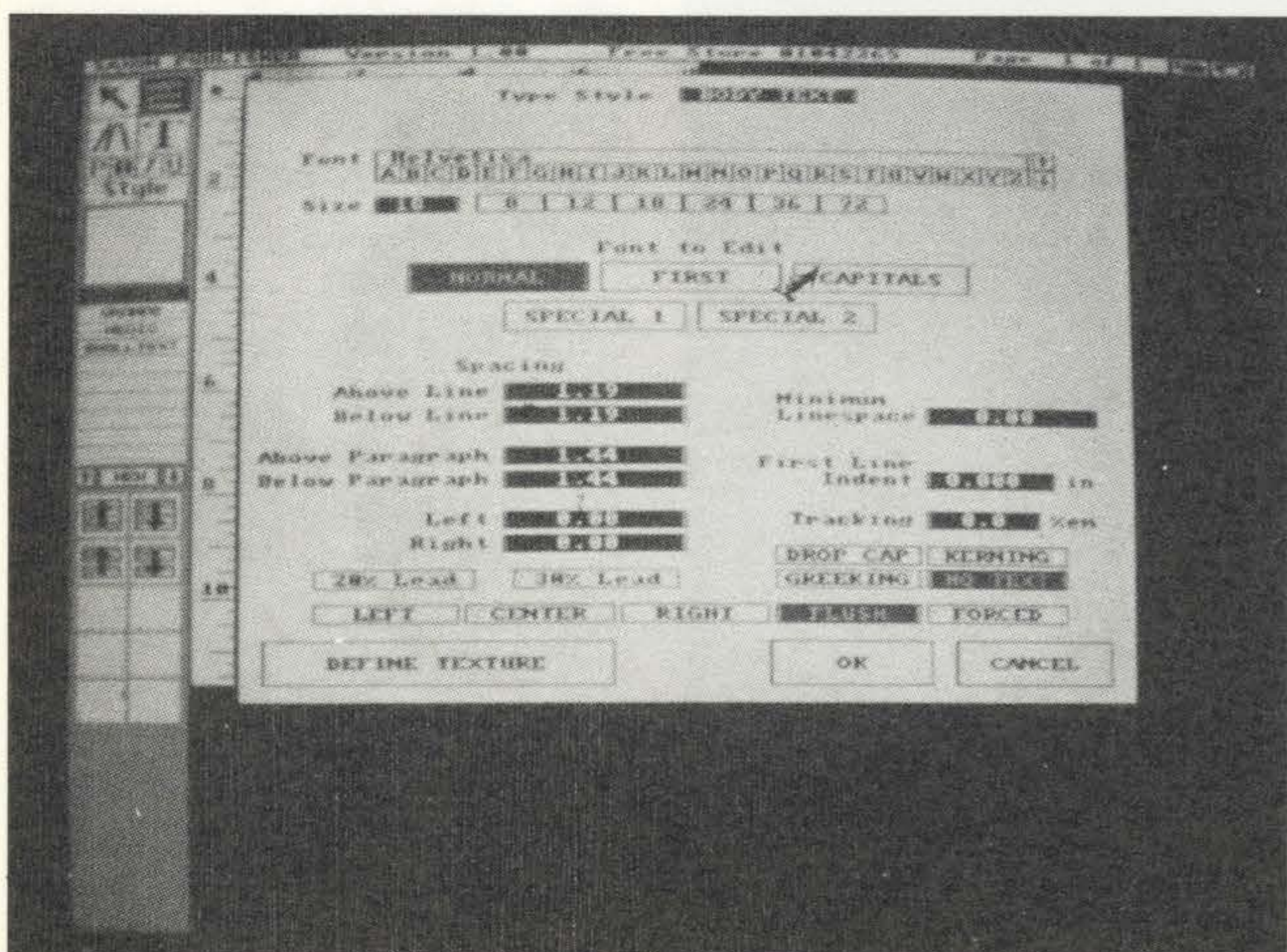
Gestione dei font con «PPM»: quelli preceduti dalla sigla CS sono i font strutturati AGFA Compugraphics.

I Tag sono etichette associate al testo che consentono di richiamare ed applicare a qualsiasi blocco parametri memorizzati relativi alla formattazione, ai font, etc.

na: gli ingrandimenti concessi sono numerosi con la possibilità di zoom. Tutte queste opzioni, sommate alla possibilità di spostare le pagine nei documenti e qualsiasi oggetto da una pagina ad un'altra, forniscono una dotazione più che buona per le operazioni descritte.

Anche con «PPM» è possibile associare ad ogni oggetto (grafico o di testo) una retinatura di fondo definendone il tipo di contorno, il tipo di riempimento, ed i colori di sfondo e di bordo. Anche se non si raggiunge il livello di sofisticazione dei texture di «Saxon», le opportunità offerte sono nella maggioranza dei casi più che sufficienti. I pattern si possono selezionare attraverso una finestra richiamata dall'**opzione Fill Style**, mentre da una finestra simile, richiamata dall'**opzione Line Style**, si può selezionare il tipo di contorno degli oggetti.

MODIFICA PARAMETRI DEI BOX

Una volta creato un box è possibile modificarne i parametri (opzione **Edit Coordinates**), e sovrapporlo ad altri box. In fase di preparazione dei box non è necessario preoccuparsi di come il testo dovrà scorrere attorno ad eventuali immagini, in quanto si interverrà successivamente sui box contenenti le figure grafiche.

«Saxon» usa estesamente i Tag, permettendo di definirne sia per il testo che per ogni altro elemento che compone una pagina.

L'unica operazione che non è possibile fare è quella di costruire box di forma non rettangolare, cosa fattibile invece con «Saxon». Insieme ai texture, la possibilità di costruire box di forma qualsiasi è una delle innovazioni principali del programma. Nella modalità cursore (icona rappresentante la freccia attivata) vengono mostrate due icone con le quali è possibile costruire box rettangolari oppure poligonali. La costruzione di un box poligonale è molto semplice in quanto, tutte le volte che si clicka sulla pagina con il mouse, viene creato un nuovo vertice del box: quando il box dispone di tutti i vertici, premendo il tasto Esc si termina l'operazione.

È possibile ridimensionare a piacere il box ridefinendone completamente la forma; ad ogni box costruito viene associata una serie di parametri, visualizzabili e modificabili attraverso l'**opzione Alter Active** del **menu Box**. Dalla finestra che questa opzione richiama si possono effettuare anche distorsioni.

Il contenuto del box in elaborazione è inclinabile lungo l'asse normale al foglio e lungo quello assiale (**comandi Baseline Elevation** e **Forward Slant**); in questo modo il testo può essere inclinato senza però inclinare anche il box, come succede con «PPM». Oltre a ciò, sono definibili la scala lungo gli assi X ed Y, i margini, la trasparenza del box rispetto alla pagina ed al testo eventualmente contenuto in un box sottostante, e la definizione del texture; per quest'ultima opzione vale tutto quello già detto per la pagina.

Come con «PPM», gli ingrandimenti disponibili sono molteplici, con possibilità di zoom definibile: la griglia di riferimento è però purtroppo misurata in frazioni di pollice. I motivi grafici, come in «PPM», sono inseribili nella pagina grazie a strumenti interni. Anche in «Saxon» sono ovviamente presenti tutti i comandi per lo spostamento delle pagine e degli oggetti, anche se con un numero di opzioni inferiore a quello offerto da «PPM». Sicuramente migliore è invece il modo di gestire i box; inoltre, a favore di «Saxon» c'è da sottolineare che attraverso i tag ed i tasti funzione, do-

Con «PPM» il testo può scorrere intorno ad un'immagine in diversi modi. Questa finestra seleziona il tipo di scorrimento.

po un po' di pratica si lavora a velocità sorprendente.

LA GESTIONE DEL TESTO

Sia «Saxon» che «PPM» permettono di inserire il testo direttamente, fornendo tutte quelle opzioni tipiche dei word processor per la copiatura, l'incollaggio e la cancellazione dei blocchi di testo. Oltre a questo, è possibile effettuare operazioni di ricerca e di sostituzione delle stringhe di testo. La velocità di editing e di visualizzazione (capita spesso che per avere una buona visione dei caratteri, il paragrafo vada fuori dalla porzione di testo visualizzata comportando continui refresh video), rendono preferibile preparare il testo con un word processor.

Attraverso il **menu External**, «Saxon» permette di importare testo in **formato ASCII** non esteso: ciò si traduce nel fatto che tutte le vocali accentate contenute nel testo importato vanno perdute. Questo, insieme al fatto che non è possibile la sillabazione, costituisce uno dei maggiori punti deboli di «Saxon». Oltre che lo standard ASCII sono riconosciuti anche i file di «**Word Perfect**».

Senz'altro superiore «PPM»: oltre che il **set ASCII** esteso (vocali accentate comprese), il programma riconosce correttamente i file dei word processor «**Word Perfect**», «**Excellence!**», «**ProWrite**» ed il **formato IFF-FTXT**. Quest'ultimo standard è uno di quelli utilizzati da «**C1-Text**», che quindi si integra perfettamente con «PPM».

Quando il testo viene importato in «Saxon», compare un tag corrispondente nella tabella posta sulla barra di comando. Per introdurlo in un box basta attivare quest'ultimo con il mouse dopo aver precedentemente selezionato il tag relativo al testo. Con «PPM» basterà invece attivare il box in cui si desidera porre il testo e poi importarlo attraverso l'**opzione Import/Text** del menù File.

«PPM» permette di utilizzare, oltre che i normali font di vario tipo ed i 12 stili presenti, anche due **font AGFA Compugraphics**: il **Tyme** e l'**Helvet**. Sono inoltre disponibili su dischi a parte altri tipi di font Compugraphics.

Questi font, utilizzati per primi dalla Gold Disk per «Professional Page», sono font strutturati che vengono ricalcolati ad ogni ingrandimento della pagina. In questo modo la velocità diminuisce, ma la qualità di visualizzazione è di gran lunga migliore che nel caso di uso dei normali font bitmap.

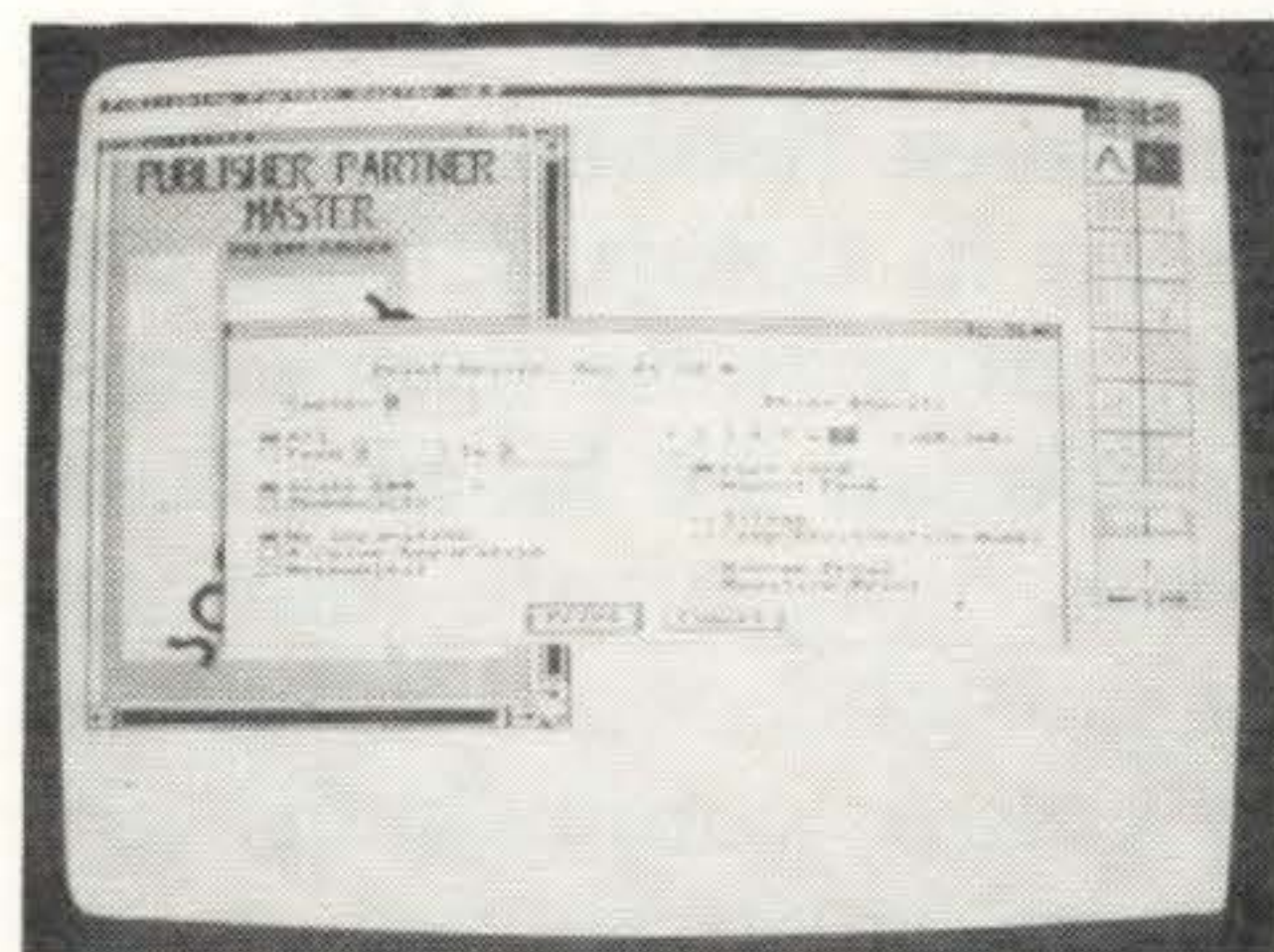
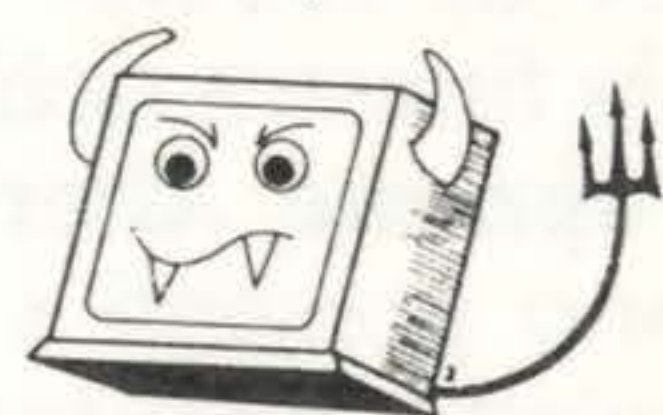
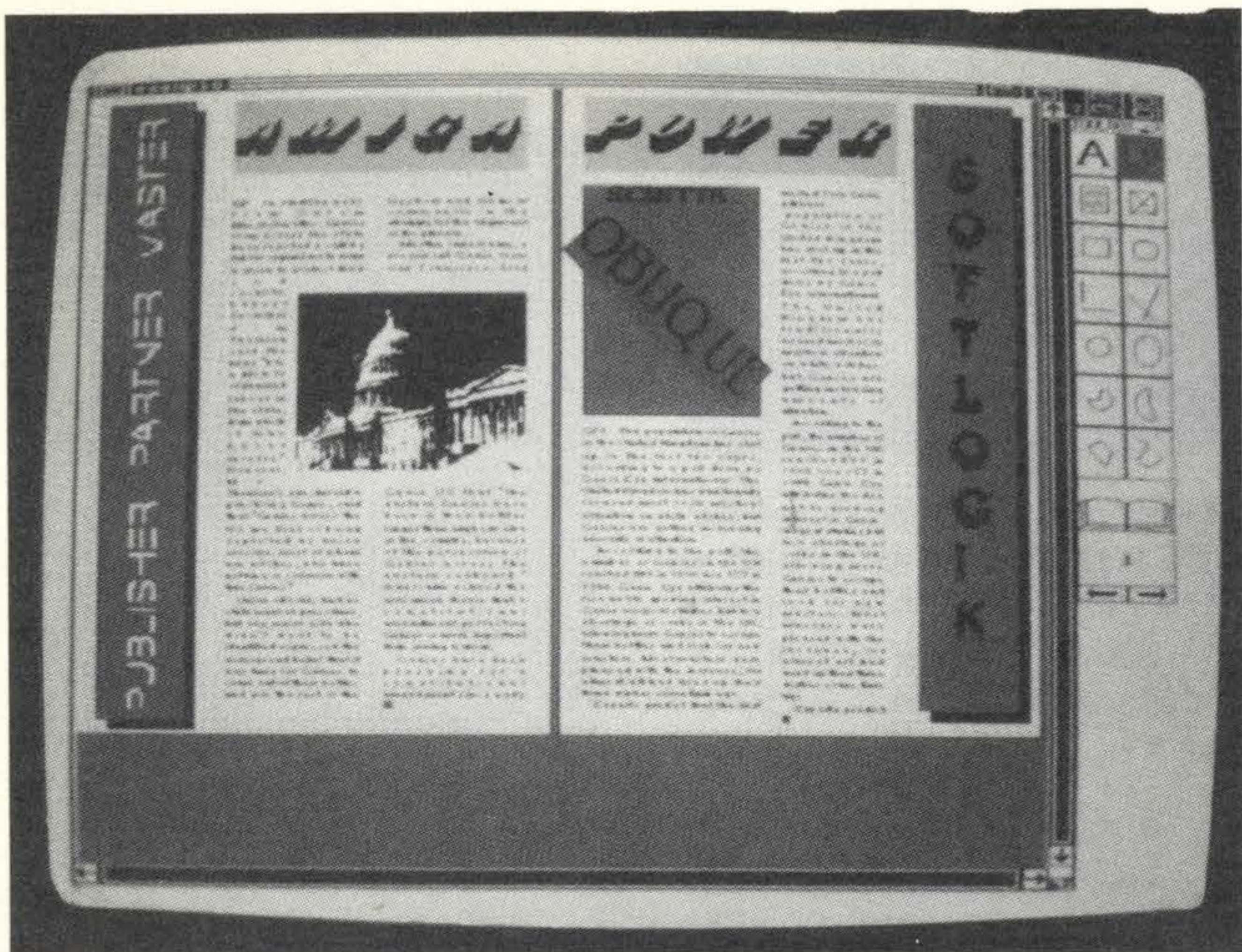
I corpi permessi da «PPM» con tutti i font sono ben 23, con misure che vanno dal minuscolo 3 all'enorme 216. Alla scelta dei font e degli stili è dedicato l'intero **menu Style**. Selezionando l'**opzione Fonts/Point** si accede ad una finestra dalla quale è possibile cambiare tipo e corpo del font. Il testo presente nella pagina è suscettibile di modifiche nel momento in cui viene evidenziato con il pointer. Sempre con questa modalità, il testo può essere formattato ed allineato a destra, a sinistra, al centro, o perfettamente giustificato. È possibile inoltre sillabare il testo definendo eventuali regole ed eccezioni nel caso si utilizzino parole di più lingue in un unico documento (ad esempio un data sheet comprendente termini tecnici inglesi o tedeschi).

I TAG DI PPM

Ma non è tutto: anche «PPM» mette a disposizione dei tag per il testo. Con un tag è possibile memorizzare tutte le informazioni circa l'impaginazione ed il font usato. Tutti i parametri usati da «PPM» vengono memorizzati permettendo di formattare anche in modo complesso un qualsiasi testo molto velocemente. La finestra che consente di definire e di utilizzare i tag viene richiamata dall'**opzione Tag** del **menu Text**. Occorre infine osservare la presenza di un'altra potentissima opzione chiamata **Search/Replace Attributes**; tramite essa, si possono scambiare tutti gli attributi di un blocco di testo, compresi i vari Tag definiti.

«Saxon» permette di manipolare i blocchi di testo in modo molto intuitivo: lo stile si può modificare direttamente in modalità testo scegliendo tra i classici neretto, sottolineato, corsivo e tondo.

Tutte le altre operazioni di impaginazione sono impartibili in modalità paragrafo; come con «PPM» si può allineare il testo a de-

I box poligonali di «Saxon»: tutti i parametri relativi ad un box sono gestibili da questa finestra, compresa la rotazione lungo due assi.

I risultati in fase di stampa sono, con «PPM», eccezionali. Il programma consente un output di ottima qualità anche su stampanti ad aghi.

Uno degli strumenti più potenti da «PPM» è la gestione di pagine destre e sinistre. È possibile anche visualizzarle affiancate.

stra, a sinistra, al centro o giustificarlo; si possono definire i tab, variare il tipo di font e la misura del corpo, ed anche iniziare un nuovo paragrafo con una lettera di dimensioni maggiori rispetto a quella del testo seguente. Questi parametri sono contenuti all'interno di una finestra richiamabile dall'**opzione Alter Current** del **menu Type Style**. Il numero di font forniti è più che sufficiente, ma le misure possibili sono solo 6: poche rispetto alle 23 di «PPM».

Si può visualizzare il testo in modalità molto definita (visualizzazione lenta) oppure in modalità veloce. In alta qualità il risultato è più che sufficiente, anche se distante dai risultati ottenibili con i font Compugraphics di «PPM». Sempre nella finestra richiamata dall'**opzione Alter Current** sono presenti i parametri relativi all'interlinea, al kerning dei caratteri all'indentazione, e così via.

Anche per il testo vale il discorso fatto per le pagine e per i box a proposito dei texture: una volta definiti tutti questi parametri è possibile memorizzarli in un tag. Basta clickare sull'**icona New** sotto la tabellina posta sulla barra dei comandi per poter scrivere in un'apposita finestra la denominazione del tag creato. Il tag appena definito verrà mostrato nella tabellina, e sarà quindi richiamabile in ogni momento.

In definitiva, «PPM» offre un numero maggiore di possibilità di manipolazione ed esce vincitore per quello che riguarda l'importazione, la sillabazione e la possibilità di avere il set ASCII esteso comprendente le vocali accentate. «Saxon» permette però un uso dei Tag più facile e veloce e l'utilizzo dei texture.

IMMAGINI E STAMPA

Entrambi i programmi consentono di importare immagini grafiche di tipo bitmap e strutturate. Questa operazione viene, con «Saxon», effettuata tramite le opzioni presenti nel **menu External**: con l'**opzione Bitmap** si possono importare immagini IFF con qualsiasi risoluzione e numero di colori. Esiste anche un'opzione denominata **Bitmap 24 bit** che consente d'importare immagini grafiche con 16 milioni di colori, come quelle generate da programmi come «**The Art Department**».

Come ad ogni altro oggetto trattato, anche all'immagine caricata viene associato un tag mostrato nella solita tabella. Sarà sufficiente creare un box e clickare sul tag relativo all'immagine importata, per vedere l'immagine riposta nel box. La visualizzazione delle immagini grafiche può essere disabilitata grazie all'**opzione Bitmap Hide Page/Display Page**: in questo modo il refresh video avverrà in minor tempo. Tutte le immagini importate verranno visualizzate in bianco e nero anche se, ovviamente, in fase di stampa verrà mantenuta la palette originale.

Le immagini strutturate importabili sono esclusivamente di due tipi: quelle create con il programma «**ProVector**», oppure in formato **Encapsulated PostScript**, compatibili con quelle generate da altri programmi di Dtp.

L'operazione di impaginazione del testo attorno alle immagini si riduce al solo tracciamento dei box. Potendo infatti disegnare box poligonali di forma qualsiasi, sarà sufficiente «sagomare» opportunamente i box per ottenere qualsiasi tipo di disposizione. Se dal punto di vista della gestione delle immagini bitmap non si può muovere nessun appunto, criticabile è invece la scelta di non supportare file strutturati del tipo «**Aegis Draw**» o «**Professional Draw**».

Le immagini strutturate importabili da «PPM» comprendono invece quelle negli standard maggiori appena citati. Le immagini IFF possono essere importate direttamente sulla pagina, oppure su di una finestra separata: in quest'ultimo modo è possibile agire più facilmente sull'immagine, manipolandola più velocemente.

L'impaginazione del testo attorno alle immagini avviene abbastanza semplicemente, utilizzando l'**opzione Text Runaround** del **menù Object** dopo aver attivato il box relativo all'immagine.

In seguito alle prove effettuate, vi è un solo appunto da muovere a «PPM»: qualche immagine nello standard «Aegis Draw» non veniva caricata correttamente, in quanto alcuni particolari, quelli molto piccoli, andavano persi oppure posizionati non correttamente. Ciò che non avviene con le immagini in formato «Professional Draw».

LA STAMPA CON SAXON E PPM

La stampa con «PPM» è attivabile selezionando l'**opzione Print** del **menu File**. Viene visualizzata una finestra contenente tutti i parametri (tra cui quelli relativi alla densità di stampa) compresi quelli che attengono alla separazione del colore. È inoltre possibile stampare in negativo e in modo speculare (**Negative Print** e **Mirror Print**).

L'uso esteso dei box poligonali in «Saxon» consente di costruire dei layout di pagina impossibili da realizzare con qualsiasi altro programma.

«Saxon» gestisce solo stampanti laser PostScript; la stampa del documento o della singola pagina avviene attraverso due sotto-opzioni del **comando Print** del **menù Document**. Anche in questo caso, tutti i parametri di stampa sono regolabili da una finestra. Tutto ciò che è stato detto per «PPM» vale anche per «Saxon», che permette anche di reindirizzare la

stampa su disco e di agire sui parametri relativi alla separazione del colore in quadricromia, nonché di variare la densità di stampa.

COME STAMPANO?

Da prove effettuate, entrambi i programmi stampano molto bene: l'uscita su stampanti ad aghi di «PPM» consente risultati notevoli, sebbene a volte alcuni motivi grafici strutturati vengano persi o mal posizionati. Dipende comunque dalla stampante posseduta e dal relativo driver perché, ad esempio, usando una Mannesman laser non PostScript la stampa è sempre stata perfetta, mentre con una Nec P6+ si sono notate occasionali imprecisioni. Questi fenomeni, comunque, avvengono raramente, e solo per oggetti molto vicini ai margini.

«Saxon» e «PPM» rappresentano dunque oggi due strumenti avanzatissimi nel settore Dtp, ed entrambi risultano essere abbastanza stabili all'atto pratico: in particolare «Saxon» dimostra un'affidabilità molto buona per essere alla prima versione.

NO AGLI «ESPERIMENTI»

L'affidabilità di «PPM» dipende invece molto dall'utente: adoperando il programma in modo corretto, avendo sempre ben presente quello che si sta facendo, essa è più che buona. Al contrario, «PPM» non gradisce gli esperimenti: giocando con le varie opzioni senza cognizione di causa si rischia il blocco di sistema anche perché, causa l'elevatissimo numero di comandi disponibili, può capitare d'impartire una sequenza di opzioni non compatibili. ☐

AMIGABYTE È LA TUA RIVISTA !

Contribuisci anche tu a renderla sempre più bella ed interessante rispondendo a qualche domanda ed aiutandoci a darti ciò che vuoi!

Quanti anni hai?..............................

Di cosa ti occupi (studio, lavoro)?..........

....

Quale Amiga possiedi?

☐ 500 ☐ 1000

☐ 2000 ☐ 3000

Quali di queste periferiche possiedi?

☐ stampante ☐ disk drive esterno ☐ hard disk

☐ modem ☐ espansione di memoria

☐ schede acceleratrici

☐ genlock o digitalizzatore video

☐ equipaggiamento midi o digitalizzatore audio

☐ schede emulazione MsDos o SideCar

☐ altro

Da quanto leggi AmigaByte?.................

....

La trovi facilmente in edicola?..............

....

Di AmigaByte segui di più:

☐ la rivista

☐ il dischetto

☐ entrambi

Sei più interessato a:

☐ recensioni utility ☐ recensioni giochi

☐ didattica ☐ altro

Quali temi preferisci?

☐ musica ☐ grafica e animazione

☐ programmazione (Basic, C, Assembler)

☐ applicativi (word processing, database, spreadsheet)

☐ giochi e avventure ☐ desktop publishing

☐ desktop video ☐ prove hardware

Quali sono le tue rubriche preferite di AmigaByte?

☐ Software Express ☐ Tips & Tricks

☐ Adventure ☐ I corsi di programmazione

☐ Tools ☐ Le recensioni/tutorial

☐ Megagame ☐ Altro

Che livello di approfondimento desideri per gli articoli?

☐ Per principianti ☐ Intermedio ☐ Per esperti

Che voto daresti in media ad AmigaByte?

Disco:		Rivista:	
	Pessimo		Pessima
	Scarso		Scarsa
	Discreto		Discreta
	Buono		Buona
	Molto buono		Molto buona
	Eccellente		Eccellente

Ti piacciono le copertine di AmigaByte?.

....

Cosa vorresti trovare sulla rivista?.........

....

Cosa vorresti trovare sul disco?...........

....

Quali altre riviste leggi?.....................

....

Hai dei suggerimenti? Quali?................

....

Nome.. **Cognome**..

Indirizzo..

Città.......................... **Prov.**..................... **C.A.P.**....................

Inserisci i tuoi dati (facoltativi) e spedisci questo tagliando, o una sua fotocopia, ad AMIGABYTE, C.so Vittorio Emanuele 15, 20122 Milano.

KickStart 2.0 per tutti!

Non è necessario un Amiga 3000 per usare il nuovo sistema operativo. Ecco come caricare KickStart e WorkBench 2.0 anche su Amiga 500 e 2000 senza bisogno di modifiche hardware o di cambiare le Rom.

di VITTORIO FERRAGUTI

La nuova versione 2.0 del sistema operativo di Amiga, le cui innovazioni sono state illustrate in dettaglio nelle pagine del fascicolo 29 di AmigaByte, viene comunemente ritenuta privilegio esclusivo dei possessori di Amiga 3000.

Allo stato attuale non è infatti ufficialmente possibile per gli utenti di Amiga 500 o 2000 effettuare un *upgrade* del proprio computer per installare in esso le Rom da 512K con la nuova versione del sistema operativo. Anche i possessori di Amiga 1000 sono impossibilitati a caricare nei loro computer un dischetto **KickStart 2.0**, poiché il nuovo sistema operativo occupa ben 512K di memoria e la ram riservata al KickStart negli Amiga 1000 ammonta a soli 256K.

Esiste tuttavia un metodo alternativo mediante il quale chiunque, indipendentemente dal modello di Amiga posseduto, può godere dei benefici del nuovo sistema operativo senza bisogno di installare nuovo Rom o di intervenire via hardware. L'unico requisito indispensabile è che nel proprio computer sia installata un'espansione di memoria da almeno **1 mega** (per un totale di **1,5 mega**), situata a partire dalla **locazione $200000**. Potete verificare la presenza di memoria a partire dalla locazione $200000 digitando da Cli il **comando "MergeMem"**, presente nella **directory "System"** del disco **WorkBench 1.3**.

Prima di procedere con la spiegazione, è bene chiarire due punti fondamentali sui quali spesso regna molta confusione tra i meno esperti.

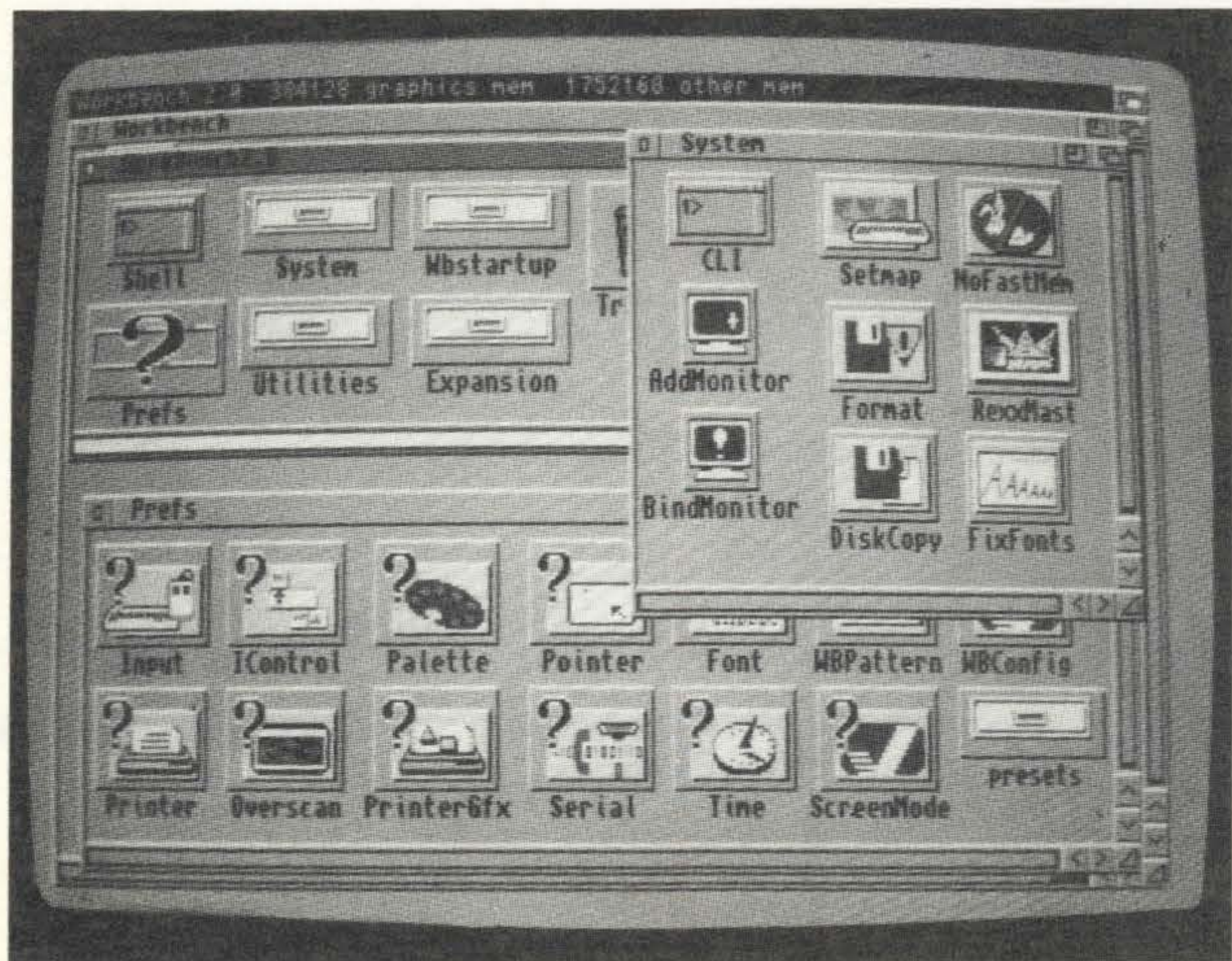
La prima importante avvertenza è che per utilizzare il KickStart 2.0 **non** è necessario possedere un Amiga 3000, né una scheda acceleratrice, né il Fatter Agnus o l'Enhanced Chip Set (ECS). Tranquillizzatevi: il nuovo KickStart può essere utilizzato senza problemi anche con un normalissimo Amiga 500 o 2000 dotato della vecchia versione del Fat Agnus.

QUALCHE CHIARIMENTO

Ovviamente il nuovo sistema operativo sfrutta, se presenti, le potenzialità offerte da hardware aggiuntivo come schede acceleratrici dotate di **MMU** (Memory Management Unit) e dai nuovi chip grafici **Denise** dell'ECS; la loro presenza non è comunque indispensabile per un corretto funzionamento del sistema in modalità 2.0.

Inoltre, per caricare in memoria il KickStart 2.0 su Amiga 500 o 2000, occorre impiegare una speciale versione del KickStart memorizzata sotto forma di «**Kickfile**», che viene fornita da Commodore solo ai propri sviluppatori autorizzati; non è quindi possibile usare il dischetto KickStart che viene fornito ai possessori di Amiga 3000, memorizzato in formato non Dos (esattamente come i dischetti KickStart usati dai possessori di Amiga 1000).

I «Kickfile» distribuiti agli sviluppatori sono in genere contrassegnati da nomi come «**Kick.303.20**», in cui le tre cifre centrali identificano la revisione del KickStart 2.0 (in questo caso 36.303), e forniti su dischetti contenenti, solitamente, anche due utility necessarie al loro caricamento in memoria.

I due programmi si chiamano «**KickIt**» e «**ZKick**», e svolgono entrambi la stessa funzione: caricare in ram il KickFile e fare in modo che Amiga utilizzi il suo codice KickStart al posto di quello presente in Rom.

```
; Questa Startup-Sequence  consente  di scegliere al momento del boot
; se caricare il KickStart 1.3 o 2.0,  e di conseguenza se utilizzare
; il WorkBench 1.3 o 2.0.  La configurazione  del sistema testata e':
; Amiga 500 con KickStart 1.3 ed HardDisk A590 con 2 mega di memoria.

VERSION >NIL: graphics.library 36

; Il comando Version restituisce il valore WARN se la versione della
; libreria grafica non e' uguale o maggiore a 36 (ovvero se il Kick-
; Start non e' il 2.0).

IF WARN

        ASK "Vuoi il KickStart 2.0 (y=si'/N=no)"

        ; Chi usa ARP puo' utilizzare questa sintassi:
        ; ASK "Vuoi il KickStart 2.0 (s/N)" "s" "n" TIMEOUT 15

        IF WARN

                ZKICK sys:WorkBench2.0/kick.303.20
                ; Se si utilizzano espansioni di memoria o hardware
                ; diverso,  occorre usare KICKIT al posto di ZKICK.
                QUIT

        ELSE

                SETPATCH >NIL: r
                Sys:System/FASTMEMFIRST >NIL:
                ASSIGN SYS: DH0:

        ENDIF

ELSE

        ASSIGN SYS: DH0:WorkBench2.0
        ASSIGN WorkBench2.0: DH0:WorkBench2.0

ENDIF

ASSIGN c: sys:c
ASSIGN s: sys:s
ASSIGN l: sys:l
ASSIGN fonts: sys:fonts
ASSIGN devs: sys:devs
ASSIGN libs: sys:libs
ASSIGN T: RAM:

VERSION >NIL: graphics.library 36

IF NOT WARN

   EXECUTE Dh0:WorkBench2.0/s/startup-sequence

ELSE

   EXECUTE Dh0:s/startup-sequence1.3

ENDIF
```

La Startup-Sequence modificata per scegliere quale KickStart e quale WorkBench caricare su di un Amiga 500 espanso.

La scelta tra l'uso di uno dei due deve essere fatta in base al tipo di configurazione del sistema: «KickIt» viene normalmente usato su Amiga dotati di schede di espansione autoconfiguranti (inserite nel primo slot di espansione, nel caso di Amiga 2000) o di schede acceleratrici come la **A2630**; «ZKick» viene usato quando la ram espansa presente nel sistema è quella montata nell'hard disk **A590** (nel caso degli Amiga 500) o sui controller per hard disk **A2091**, o sulle schede di produzione **GVP** (nel caso degli Amiga 2000). Il loro funzionamento è comunque pressochè identico.

La reperibilità del nuovo KickStart potrebbe rappresentare invece un problema: tuttavia, sebbene si tratti di materiale riservato in esclusiva agli sviluppatori Commodore, i «Kickfile» circolano in grande quantità nel circuito sotterraneo degli utenti Amiga più smaliziati, e si riescono ormai a trovare con discreta facilità.

L'INSTALLAZIONE

La procedura che stiamo per descrivere è stata messa a punto su di un Amiga 500 dotato di KickStart 1.3 e del vecchio Fat Agnus, con un hard disk A590 e 2 mega di memoria Ram montata sul controller del disco rigido. Esistono ovviamente anche altri sistemi per procedere all'installazione: è possibile ad esempio creare due partizioni distinte sull'hard disk per contenere i file relativi ai due sistemi operativi e renderne più efficiente il caricamento.

Il metodo da noi illustrato è però il più semplice per chi non desideri effettuare nuove partizioni sul disco, e non voglia mettere mano alla *Mountlist.*

Naturalmente anche chi non possiede un hard disk può comunque dare il via al caricamento del nuovo sistema operativo; la procedura per lui sarà solo più macchinosa, dovendo caricare da dischetto il KickStart 2.0 e successivamente inserire il floppy corrispondente al WorkBench adatto.

Passiamo all'azione: effettuiamo il boot dal nostro hard disk come di consueto e, dopo essere entrati in ambiente CLI o Shell, creiamo una nuova directory chiamata «WorkBench 2.0», nella quale copieremo l'intero contenuto di un dischetto WorkBench2.0 standard (potete farvelo prestare da un amico possessore di Amiga 3000).

Per eseguire queste operazioni, è sufficiente digitare questi comandi, dopo aver inserito il dischetto del WorkBench2.0 nel drive interno di Amiga:

Makedir Sys:WorkBench2.0
Copy Df0:#? Sys:WorkBench2.0 all

Nella stessa directory copiate anche l'ultima versione del Kickfile in vostro possesso, mentre copierete nella directory «c:» del disco rigido il file «ZKick» o «KickIt» (la scelta è da effettuarsi in relazione alla configurazione del sistema).

Inseriamo quindi nel drive interno di Amiga il dischetto contenente il Kickfile e lo «ZKick» (o il «KickIt») e digitiamo questi comandi:

Copy Df0:Kick.303.20 Sys:WorkBench2.0
Copy Df0:Tools/Zkick c:

L'esempio appena riportato è valido nel caso il file «ZKick» si trovi in una directory chiamata «**Tools**» del dischetto contenente il «Kickfile» (nella maggior parte dei casi è così). In caso contrario, occorre modificare di conseguenza la sintassi del comando.

Ora dobbiamo fare una copia della Startup-Sequence del vostro hard disk, che verrà usata per lanciare il sistema in modalità KickStart 1.3:

Copy S:Startup-Sequence S:Startup-Sequence1.3

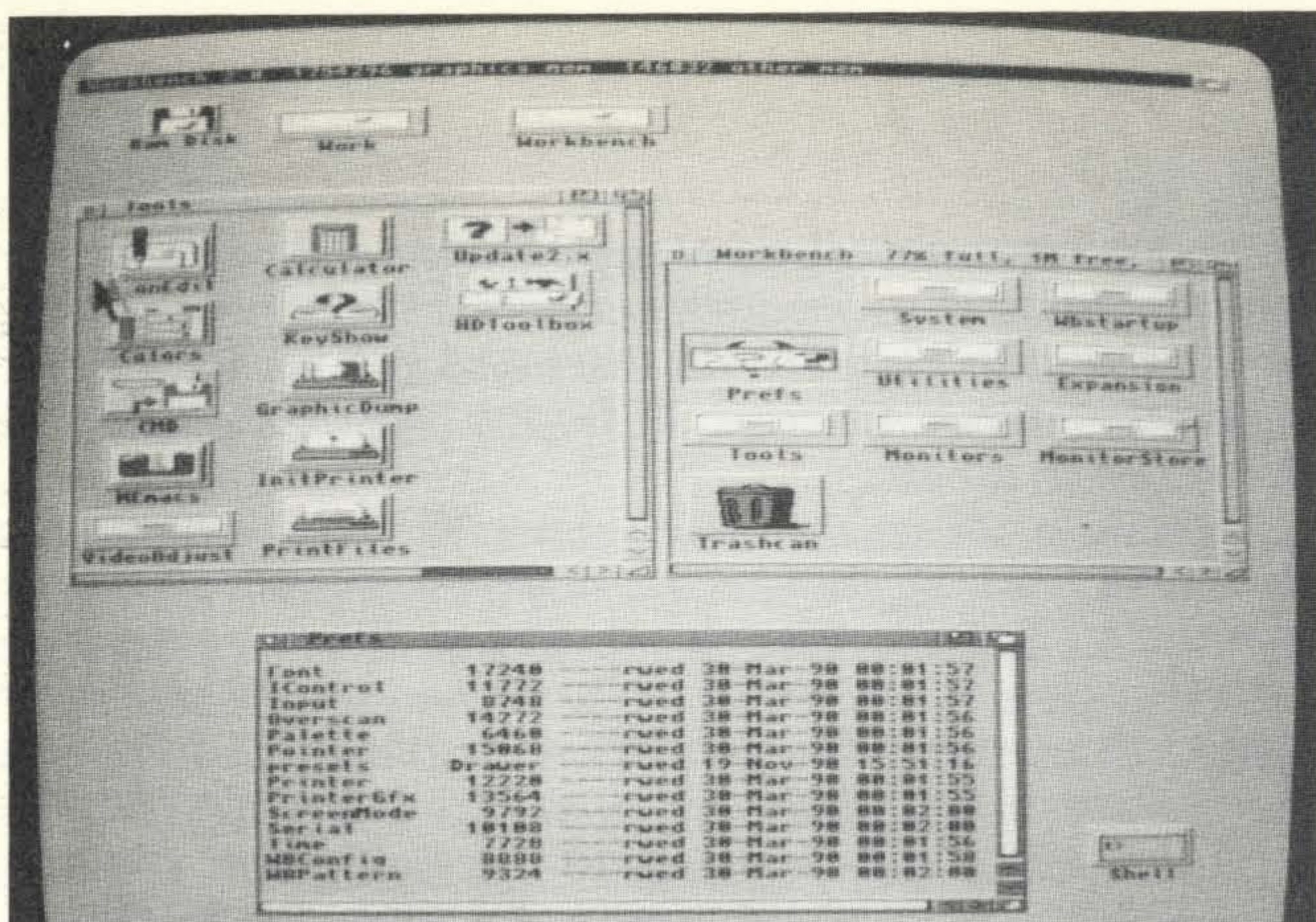
Cancelliamo la vecchia versione della Startup-Sequence ed inseriamo quella nuova, con la quale potremo attivare la versione desiderata del KickStart ad ogni avviamento del sistema:

Delete S:Startup-Sequence
Ed S:Startup-Sequence

Nel caso ci accorgessimo di avere commesso errori o cambiassimo idea, potremo sempre richiamare la vecchia Startup-Sequence poiché l'abbiamo copiata precedentemente nel file «**Startup-Sequence1.3**».

A questo punto non resta altro da fare che modificare la Startup-Sequence ed implementare la selezione del KickStart al momento del boot del sistema digitando, tramite l'**editor «Ed»** appena richiamato, i comandi che stiamo per descrivere.

UNA STARTUP PER TUTTI I GUSTI

La prima cosa da fare, nella Startup-Sequence, è controllare quale versione del KickStart è attiva. Dal momento che la Startup-Sequence è inizialmente una sola, indipendentemente dal KickStart usato, occorre sapere

qual è il KickStart attivo dopo il boot per poterci regolare di conseguenza:

Version >NIL: graphics.library 36

Il comando «Version» va a controllare il numero di versione della libreria grafica (presente nel KickStart). Se il risultato è inferiore a 36, la versione del KickStart è la 1.3, altrimenti è la 2.0. Nel primo caso, viene generato un codice di errore 5 (ovvero Warn), che viene controllato dal seguente ciclo If...Else...Endif:

```
If Warn
    ask "Vuoi il KickStart 2.0 (y=si'/N=no)"
```

Se stiamo usando il KickStart 1.3 (ovvero se il precedente controllo ha generato un codice di errore Warn), viene chiesto se vogliamo caricare il KickStart 2.0. Rispondendo Y viene generato un nuovo Warn, controllato dal seguente ciclo If...Else...Endif:

```
If Warn
    ZKick Sys:WorkBench2.0/Kick.303.20
    Quit
```

Se abbiamo risposto Yes alla domanda precedente, viene richiamato ZKick (o KickIt, a seconda della vostra configurazione di sistema) e viene caricato il KickStart 2.0; dopo qualche istante Amiga si resetterà e ripartirà in modalità 2.0.

```
Else
    SetPatch >NIL: r
    Sys:System/FastMemFirst >NIL:
    Assign SYS: dh0:
Endif
```

In caso contrario, cioè se abbiamo invece risposto No alla domanda precedente, vengono invece caricati **SetPatch** e **FastMemFirst**. Questi due comandi fanno parte della normale sequenza di avviamento del WorkBench 1.3 e dovrebbero essere già presenti nella vostra solita Startup-Sequence.

```
Else
    Assign SYS: dh0:workbench2.0
    Assign WorkBench2.0: dh0:WorkBench2.0
Endif
```

I comandi riportati sopra rappresentano l'alternativa al primo ciclo If...Else...Endif (quello controllato dal comando «Version») e vengono eseguiti nel caso il boot sia stato effettuato in modalità 2.0.

Se il KickStart è il 2.0, occorre infatti che sia caricato anche il corrispondente WorkBench 2.0 e non il solito 1.3; vengono pertanto eseguiti appositi assegnamenti, in modo che la directory contenente il disco WorkBench2.0 precedentemente copiato venga impostata come disco di sistema.

```
Assign c: sys:c
Assign s: sys:s
Assign 1: sys:1
Assign fonts: sys:fonts
Assign devs: sys:devs
Assign libs: sys:libs
```

Grazie ai vari cicli If...Else...Endif abbiamo fatto in modo da assegnare il volume di sistema **SYS:** al disco **Dh0:** (se è attivo il KickStart 1.3) o alla directory **Dh0:workBench2.0** (se è attivo il KickStart 2.0). Le righe sopra assegnano di conseguenza le varie directory di sistema (c:, 1:, Libs: ecc.) in modo che siano adeguate al WorkBench da utilizzare. Potete aggiungere alle righe sopra eventuali altri comandi «**Assign**» o «**Path**», per stabilire gli assegnamenti che devono essere attivi indipendentemente dal WorkBench usato.

A questo punto siamo pronti per caricare il WorkBench adatto e per attivare la relativa Startup-Sequence:

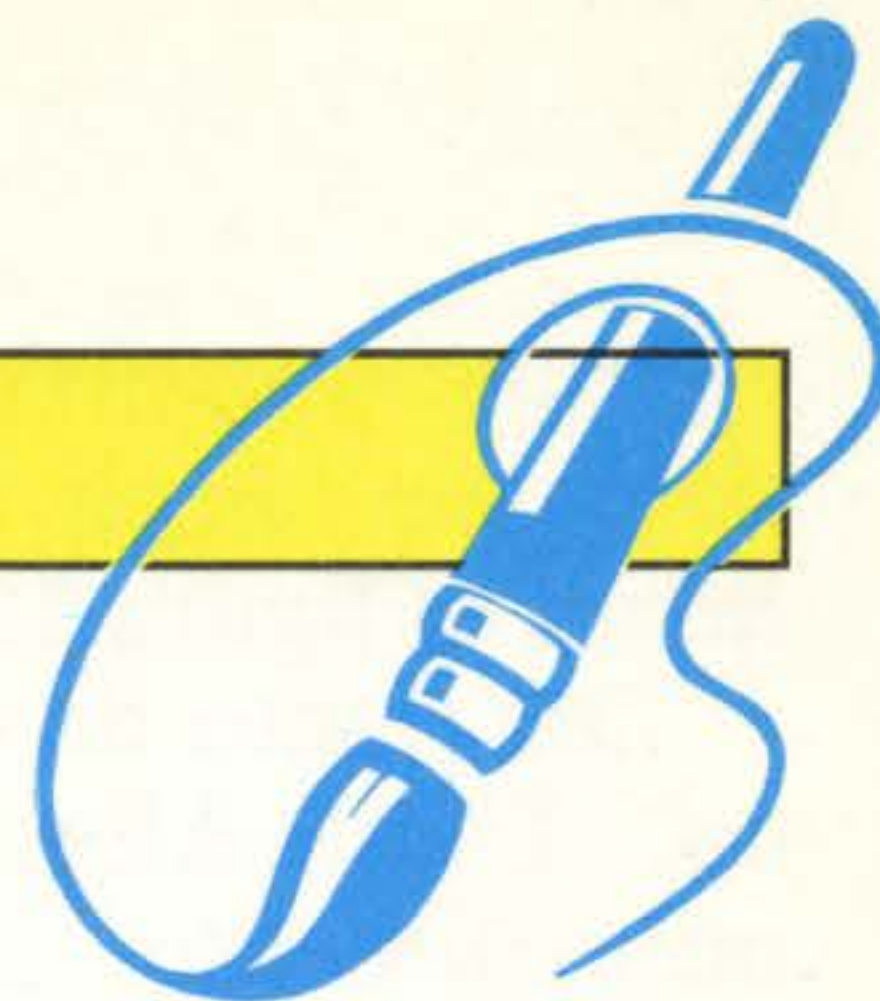
```
version >NIL: graphics.library 36
If not warn
    Execute Dh0:WorkBench2.0/s/startup-sequence
Else
    Execute Dh0:s/startup-sequence1.3
Endif
```

Con il comando «**Version**» abbiamo nuovamente controllato quale KickStart è attivo; il ciclo If...Else...Endif successivo lancia la Startup-Sequence corrispondente al WorkBench corretto. Nel primo caso (Kickstart 2.0), viene eseguita la Startup-Sequence presente nella **directory S** della **directory WorkBench2.0**, quella che appunto contiene il dischetto WorkBench2.0 precedentemente copiato; nel secondo caso (KickStart 1.3) viene eseguita la nostra solita Startup-Sequence, quella stessa che all'inizio abbiamo copiato nel file «**Startup-Sequence1.3**». Da quest'ultimo file è bene rimuovere tutte le righe già eseguite nella Startup-Sequence appena digitata, ovvero quelle relative ai comandi «**Setpatch**», «**FastMemFirst**» ed i vari «**Assign**» e «**Path**» già impostati qui.

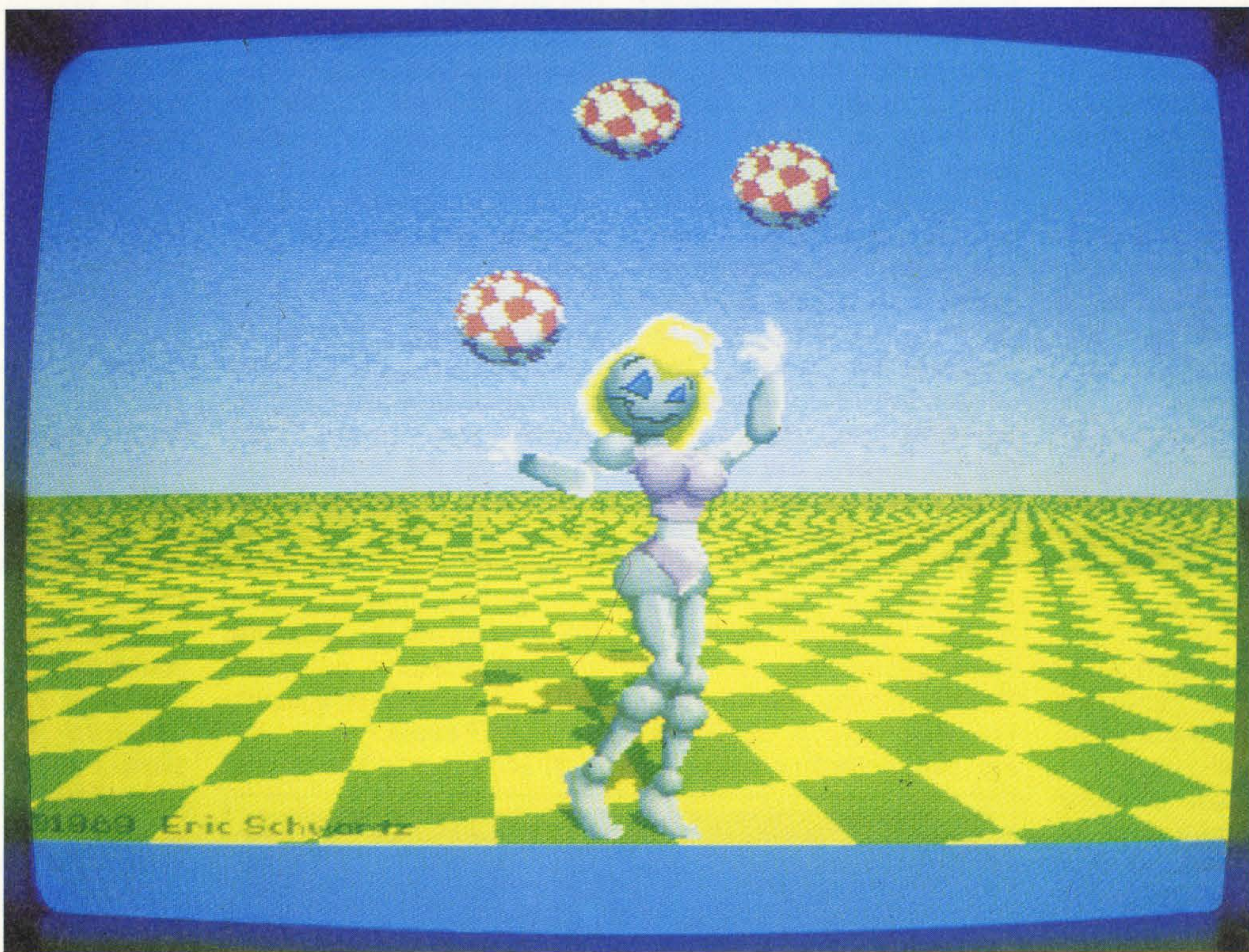
IL GIOCO E FATTO

Dopo aver controllato di non aver commesso errori di digitazione, possiamo salvare la Startup-Sequence appena inserita, uscendo dall'editor Ed premendo **Esc** seguito da **X** e da **Return**.

A questo punto potete resettare Amiga e dopo qualche istante vi apparirà la domanda «Vuoi caricare il KickStart 2.0»?

Rispondendo «N» o battendo semplicemente Return, la domanda verrà ignorata e sarà caricata ed eseguita la vostra solita Startup-Sequence in modalità 1.3; rispondendo «Y», il computer dopo qualche istante si resetterà e, salvo errori, effettuerà un secondo boot con il WorkBench 2.0. □

Deluxe Paint III all'opera

Prosegue il nostro tutorial su come realizzare animazioni stupefacenti con «Deluxe Paint III». Con poca fatica? Beh, non proprio...

di LUCA MIRABELLI

Superato lo shoc dell'impatto, nel precedente fascicolo di AmigaByte, con il meraviglioso mondo delle animazioni computerizzate, vediamo ora di analizzare in dettaglio le varie potenzialità di «**Deluxe Paint III**» in questo campo; ciò che ci interessa è naturalmente contenuto nel menu intitolato **Anim**.

Le prime due opzioni non dovrebbero mettere in difficoltà nessuno, trattandosi di **Save** e **Load**. La terza, **Move**, è stata esaminata nella puntata precedente: la sua funzione è quella di sovrapporre un brush in movimento a dei fotogrammi già esistenti. Potrà esservi utile sapere che alcuni parametri relativi all'opzione Move possono essere modificati con i comandi di un altro menu, e precisamente il sottomenu **Effects Perspective**. I due più importanti sono:

— il punto di fuga della prospettiva, che si determina con l'opzione **Center**;

— la sfumatura dei contorni, detta anche **Anti-Alias**, da attivare tramite il pannello **Settings**. L'Anti-Alias permette di evitare quelle «seghettature» che caratterizzano le linee oblique, specialmente quando si lavora in bassa risoluzione. Tecnicamente, esse vengono mascherate con l'aggiunta nei punti strategici di pixel di colore intermedio tra quello dell'oggetto e quello dello sfondo. L'Anti-Alias può essere attivato a due livelli, **Low** e **High**: le differenze tra que-

ste due modalità sono evidenziate negli ingrandimenti di figura 1.

Dal punto di vista pratico, l'Anti-Alias migliora notevolmente la qualità dell'animazione, ma aumenta smodatamente i tempi di calcolo della prospettiva, e quindi quelli di esecuzione del comando Move. Prima di attivarlo, quindi, occorre essere ben sicuri che:

— il brush attualmente selezionato includa, oltre che l'oggetto da muovere, un bordo di almeno due pixel;
— la palette contenga almeno un colore intermedio da usare per la sfumatura;
— non vi siano più modifiche da apportare né all'oggetto né allo sfondo.

Sempre a proposito della Move, ecco un altro consiglio su come evitare che un oggetto che si ingrandisce muovendosi verso l'osservatore, perda di definizione dando origine a risultati come quelli in alto nella figura 6: se è possibile, è bene seguire il procedimento contrario, ovvero partire dalla posizione più ravvicinata e farlo rimpicciolire. Talvolta, però, questo non è possibile: si pensi, per esempio, ad un oggetto che entra da fuori campo.

In questo caso, procedete come segue:

— disegnate (magari sulla **Scratch Page**) l'oggetto da muovere, in dimensioni doppie del normale;
— selezionatelo come brush, e dimezzatene le dimensioni (con **Brush Size Halve**, oppure il **tasto H**). A questo punto, «DPaint» avrà in memoria sia la versione ingrandita che quella dimezzata dell'oggetto e, in caso di necessità di ingrandimenti, verrà utilizzata la prima. Il risultato è visibile nel secondo esempio della figura 6.

Fig. 1. L'Anti-Alias elimina il fastidioso aspetto «a scalini» delle linee diagonali, aggiungendo pixel colorati opportunamente.

NASCITA E MORTE DI UN FOTOGRAMMA

La voce successiva del **menu Anim**, denominata **Frames**, dà accesso ad un sottomenu composto da cinque voci. **Add Frame** inserisce un fotogramma in posizione immediatamente successiva a quella attuale, mentre **Set #** inserirà quanti fotogrammi sono necessari per giungere ad un totale che verrà introdotto da tastiera. Questa opzione è comunemente usata per iniziare un nuovo lavoro. Sia su Add Frame che su Set # c'è da dire che i fotogrammi inseriti non saranno vuoti, ma uguali a quello attualmente visualizzato. Se si tiene presente la struttura dei file Anim, risulta chiaro come per il computer sia più semplice agire in questo modo.

Copy to all ricopia il fotogramma attualmente visualizzato su tutti gli altri in memoria (non usate questo comando con leggerezza: gli altri fotogrammi saranno irrimediabilmente perduti).

Delete Frame e **Delete All** cancellano rispettivamente il fotogramma attualmente visualizzato, oppure l'intera animazione in memoria. Entrambi agiscono solo dietro conferma, e la cancellazione non è una semplice pulizia dello schermo: il fotogramma in questione cessa di esistere e, volendolo, occorrerà reinserirlo con **Add Frame**.

Il sottomenu successivo è denominato **Control**. La voce **Set Rate** permette di determinare la velocità alla quale la nostra animazione sarà eseguita, in fotogrammi al secondo: tenete presente che un valore pari a 12 fornisce già ottimi risultati, e che valori alti possono essere al di là della capacità del processore. In questo caso, la velocità effettiva sarà la massima consentita dalle caratteristiche tecniche del computer. La soglia al di là della quale questo si verifica non è costante, ma dipende dalla complessità della nostra animazione (risoluzione, numero di colori, variazioni tra un fotogramma e l'altro).

In generale, in modo **Compressed** (vedere più avanti), la velocità massima è minore che in modo **Expanded**.

Fig. 2. Lo schema di funzionamento di un Anim Brush. Muovendosi sullo schermo, esso lascia una scia di immagini in sequenza.

Previous e **Next**, equivalenti ai tasti «1» e «2», ci portano al fotogramma precedente o a quello successivo, mentre con **Go to** (tasto «3») potremo posizionarci direttamente sul fotogramma desiderato. A causa della struttura dei file Anim, comunque, tutti quelli intermedi dovranno essere calcolati dal computer.

I tre comandi rimanenti servono per eseguire l'animazione: **Play** lo farà ciclicamente; **Play Once** soltanto una volta; **Ping-Pong** dal primo all'ultimo fotogramma, poi dall'ultimo al primo, poi nuovamente dal primo all'ultimo, e così via. Brilla per la sua assenza un'opzione Play in senso inverso, della quale spesso si sente l'esigenza.

Saltiamo per un attimo il sottomenu **Animbrush**, per esaminare quello successivo, **Method**. Esso consta di due sole voci: **Compressed** ed **Expanded**, che attivano e disattivano rispettivamente la compressione dei fotogrammi; questa scelta deve essere fatta prima di generare l'animazione.

E veniamo ora ad una delle più grandi novità introdotte da «DPaint III»: gli **Anim Brush**.

ANIMARE PICCOLI OGGETTI

Detto in poche parole, un Anim Brush sta ad un brush ordinario come un'animazione sta ad un'immagine. Si tratta quindi di animazioni di dimensioni ridotte, che vengono considerate dal programma alla stregua di «pennelli» che modificano la loro forma (passando al fotogramma successivo) ogni volta che vengono utilizzati.

Per chiarire il loro funzionamento, generate tre fotogrammi. Sempre nella stessa posizione, ed all'incirca con le stesse dimensioni, disegnate nel primo fotogramma il numero «1», nel secondo il numero «2» e nel terzo il numero «3», in colori differenti. Ora selezionate **Anim Brush Pick Up** e, come se voleste prendere un brush ordinario, selezionate la parte di schermo in cui avete disegnato le vostre figure.

A questo punto potete cancellare la vostra minianimazione (**Frames Delete All**), e divertirvi con l'Anim Brush che avete appena definito. Se tenendo premuto il pulsante sinistro del mouse muovete il puntatore sulla pagina grafica, vi lascerete dietro una «scia» in cui si ripete all'infinito la sequenza 1-2-3 (Figura 2).

Vi sono alcuni parametri relativi agli Anim Brush che è possibile modificare a piacere: per farlo, selezionate la voce **Settings** del sottomenu.

Il riquadro che compare contiene, nella pagina riga, il numero di fotogrammi (cels) da cui è composto il nostro Anim Brush: tale numero non è modificabile, ed è uguale al numero di fotogrammi dell'animazione al momento del Pick Up.

La casella **Duration** indica il numero di utilizzi in cui i fotogrammi dell'Anim Brush si devono esaurire. Un esempio chiarirà tutto: con il nostro brush 1-2-3, definendo una Duration pari a 6, ogni fotogramma sarà impiegato per due volte.

La casella **Current** indica quale dei fotogrammi sarà il prossimo ad essere impiegato, mentre i tre gadget sottostanti, contrassegnati da frecce, ci permettono di definire il verso dell'animazione: dal primo fotogramma all'ultimo, dall'ultimo al primo, o in modo «ping-pong».

NUOVE POSSIBILITÀ

A questo punto la disponibilità di RAM è l'unico fattore che può limitare la vostra fantasia: in combinazione con il riquadro **Move**, gli Anim Brush permettono di ottenere risultati dal look decisamente «professionale». A titolo di esempio, in figura 3 vediamo l'imitazione di una familiare sigla televisiva: la terra che gira è un Anim Brush tratto da una libreria della «Kara Computer Graphics», i caratteri sono

Fig. 3. Il pianeta Terra rotante in questa animazione ispirata alla sigla del TeleGiornale è un Anim Brush preso da una libreria commerciale.

Fig. 4. Un'animazione tridimensionale basata sulla funzione trigonometrica del seno. Il risultato è simile a quello di una bandiera che sventola.

Fig. 5. Funzioni più complesse producono risultati esteticamente più impressionanti. L'effetto «onda» di quest'animazione è molto suggestivo.

realizzati ed animati con «DPaint».

Terminiamo l'analisi del sottomenu con le opzioni **Use** e **Free**, utili rispettivamente per riprendere «in mano» l'Anim Brush (che viene conservato in memoria anche se in seguito selezioniamo un brush ordinario), e per toglierselo di torno in maniera definitiva, liberando preziosi Kbytes.

Sotto la voce **Mode** troviamo invece le opzioni **Compressed** ed **Expanded**, per attivare o disattivare la compressione dei fotogrammi non visualizzati; occorre fare la propria scelta prima della creazione dei fotogrammi, ed in seguito non sarà più possibile modificarla.

La completa padronanza di «Deluxe Paint» è il primo obiettivo da raggiungere per chi voglia lavorare seriamente con le animazioni; in seguito, si potrà passare a programmi più complessi per lavorare con oggetti tridimensionali e magari in ray-tracing («Sculpt-Animate 4D», «Turbo Silver», «Real 3D», tanto per non far nomi), o per aggiungere effetti video («Animagic»).

Fig. 6. Un trucco per evitare che l'ingrandimento di un oggetto in movimento si trasformi in una perdita di definizione.

A PROPOSITO DI DPAINT

Approfittiamo dell'occasione per rivolgerci all'Electronic Arts, e al programmatore Dan Silva in particolare: con tutti gli Amiga 3000 e le schede acceleratrici ormai in circolazione, a quando un bel «DPaint IV» capace di sfruttare il coprocessore matematico? Questo ridurrebbe i tempi di attesa per il calcolo delle prospettive a pochi istanti!

Occorre inoltre segnalare a chi non lo sapesse che la gestione della memoria per quanto riguarda le animazioni non è certamente un punto di forza di questo programma, che spesso non libera quella che aveva allocato, conducendo presto il sistema ad un blocco da mancanza di RAM: questo difetto si manifesta soprattutto lavorando con gli Anim Brush. Inoltre, quando compare il fatidico messaggio «Not enough memory to save changes» è generalmente troppo tardi per correre ai ripari, in quanto metà dei fotogrammi dell'animazione saranno già irrimediabilmente compromessi.

Conviene quindi, periodicamente, salvare la propria animazione e l'eventuale Anim Brush in memoria, resettare il computer e ricaricare tutto: questo può portare via un po' di tempo, ma permette di evitare brutte sorprese. E, sempre per evitare brutte sorprese, tenete d'occhio (con Ctrl-A oppure **Project / About**) la quantità di memoria libera!

PER CHI NON SA DISEGNARE...

E veniamo ora all'argomento promesso la volta precedente: le animazioni di funzioni matematiche. Si tratta di animazioni con un fascino tutto «informatico» (avete presente le sigle di «Quark»?), che richiedono soltanto un po' di pazienza ed un pizzico di conoscenze matematiche per essere realizzate.

Unico strumento necessario, oltre che «Deluxe Paint», è un programma per il disegno di grafici di funzione come «**Doug's Math Aquarium**», «**MathAmation**» o «**Formula!**», quest'ultimo incluso nel dischetto di Amiga Byte 15 (ottobre 1989).

In particolare, lavorando con «Formula!», sarà necessario munirsi di una utility come «**GrabbIt**» o «**HermIt**», in quanto non è prevista un'opzione di salvataggio IFF delle immagini generate.

I possessori del programma «**Waves**», presentato sul fascicolo 29 di AmigaByte, potranno invece tranquillamente saltare i paragrafi successivi: quell'utility infatti effettua automaticamente tutti i calcoli ed i salvataggi necessari per la realizzazione di animazioni basate su funzioni matematiche.

Cominciamo dunque con la semplice oscillazione di una bandiera al vento. Dal punto di vista matematico, una bandiera ondulata può essere definita come la funzione

$$z=f(x,y)=\sin(x).$$

La prima cosa da fare è tracciarla, prendendo un intervallo qualsiasi per la y, ed un intervallo [−6+6] per la x. Il risultato sarà qualcosa di simile alla figura 4.

È questo il momento di settare tutti i parametri relativi all'animazione: prospettiva o assonometria, punto di vista, risoluzione, e così via.

In particolare, se il grafico fosse quasi piatto (può accadere con «Doug's

Math Aquarium»), moltiplicate la funzione per una costante: 5 dovrebbe bastare.

Quando sarete soddisfatti dell'aspetto della «bandiera», salvate il risultato come «**Fotogramma01**», usando «GrabbIt» se necessario.

È giunto il momento di decidere da quanti fotogrammi sarà composta l'animazione che stiamo creando. Generalmente, con venti si ottengono buoni risultati.

La nostra animazione dovrà essere ciclica: ora, poiché la funzione sin(x) è periodica di periodo 2π, aggiungendo all'argomento della funzione un valore di 2π/20 (cioè π/10) per ogni fotogramma, dopo esattamente 20 fotogrammi avremo completato il periodo della funzione ed il grafico sarà identico a quello iniziale.

Tracciate ora il grafico della seguente

f(x,y)=sin(x+pi/10)

e salvatelo come «**Fotogramma02**». Proseguite con

f(x,y)=sin(x+2*pi/10)
[Fotogramma03]
f(x,y)=sin(x+3*pi/10)
[Fotogramma04]

e così via fino a

f(x,y)=sin(x+19*pi/10)
[Fotogramma20

NOTA: «Se il vostro programma non supportasse «pi» come costante, sostituitela con il suo valore approssimato 3.14159.

ASSEMBLAGGIO DEI FOTOGRAMMI

Uscite ora dal programma di matematica, e caricate «Deluxe Paint». Definite un numero di fotogrammi pari a 20, e caricate (con **Project / Load**) il Fotogramma01.

Spostatevi (con il **tasto 2**) sul prossimo, e caricate il Fotogramma02, e così via fino al Fotogramma20.

NOTA: Se compare il messaggio «Cannot load different sized picture as Anim Frame» (causato dal fatto che alcuni programmi di matematica salvano i grafici in NTSC anziché in PAL), due sono le strade per risolvere il problema:
– caricare l'intera immagine come Brush e trasferirla sul fotogramma (solo se avete molta Chip RAM libera);
– spostarsi (con **Project Spare Swap**) sulla Scratch Page, caricare da lì l'immagine, e trasferirla nell'animazione con **Project Spare Copy to Spare**.

Salvate ora l'intera animazione con **Anim Save**, e premete il **tasto 4** per ammirare il risultato di tanti sforzi.

Se avete assimilato il meccanismo, provate quest'altra funzione:

sin(x*x+y*y+*npi/10)/**
(x*x+y*y+1)

sempre in 20 fotogrammi, facendo variare il parametro *n* da 19 a 0.

Il risultato sarà simile a quello che appare in figura 5: onde sinusoidali concentriche, come quelle generate da un sasso gettato in uno specchio d'acqua.

POTEVAMO STUPIRVI CON EFFETTI SPECIALI

In entrambi i casi precedenti, l'animazione è basata sulla periodicità della funzione. Vi sono comunque molti altri «effetti» mediante i quali è possibile animare una qualsiasi funzione di proprio gradimento:

– «**Rovesciamento**», ottenuto moltiplicando la funzione per un parametro che varia da −1 a +1 (da utilizzare con l'opzione **Play Ping-Pong**);

– «**Pulsazione**», ottenuta moltiplicando la funzione per il parametro cos(n), con n che varia da 0 a π;

– «**Altalena**»: moltiplicare la funzione per (x*n), con n che varia −.5 a +.5;

– «**Zoom in**», ottenuta aumentando Xmin e Ymin e diminuendo Xmax e Ymax di una stessa quantità per ogni fotogramma.

– «**Colpo di vento**»: moltiplicare la funzione per il

Fig. 7. In questa funzione il colore dell'oggetto cambia proporzionalmente al valore di una variabile. L'immagine è stata realizzata con un listato Basic.

parametro sin(x)/(abs(x-n)+1), con n che varia da Xmin a Xmax.

Altre modifiche possibili includono: lo spostamento (magari circolare) del punto di vista, la traslazione degli assi coordinati (che è stata utilizzata per l'animazione della bandiera), e tutto ciò che la vostra fantasia potrà suggerirvi.

Non dimentichiamoci, però, che il nostro obiettivo non è matematico, ma squisitamente estetico: potremo perciò rielaborare graficamente i singoli fotogrammi per variare l'effetto finale.

In particolare, dovreste prendere in considerazione la possibilità di variare i colori (dall'interno del programma matematico se questo lo permette, altrimenti con il solito «DPaint»), così da ottenere risultati di grande effetto come quelli di figura 7. L'algoritmo scelto, in questo caso, attribuisce un colore (che varia proporzionalmente) al valore della z in quel punto. Vi sono comunque numerose altre possibilità: a titolo di esempio indicheremo l'algoritmo utilizzato in ray-tracing, che attribuisce colori differenti a seconda della «pendenza» della funzione nel punto in esame (ovvero, per chi se ne intende, della derivata della funzione rispetto alla x oppure alla y). Anche l'attribuzione casuale dei colori può fornire risultati interessanti.

Il brutto è che solitamente queste procedure non sono implementate nei programmi di matematica, ed occorre quindi procedere con «DPaint», ad occhio o armati di una calcolatrice scientifica.

Due parole riguardo alla scelta della funzione: generalmente si ottengono risultati gradevoli con le trigonometriche seno e coseno, nonché con le radici quadrate. Sconsigliate, invece, funzioni esponenziali, polinomi di grado elevato e la tangente trigonometrica.

La funzione dovrà naturalmente essere parametrica, in modo da far variare il parametro tra un fotogramma e l'altro. In presenza di funzioni periodiche (come le trigonometriche) si ottengono effetti molto gradevoli aggiungendo ai loro argomenti un parametro **n/p**, dove **n** è il numero del fotogramma da generare e **p** il perido della funzione.

Una volta compreso il meccanismo fondamentale, l'animazione di funzioni non presenta difficoltà di sorta, e richiede unicamente un po' di pazienza. Se si conosce la matematica si potrà partire da una forma per giungere alla funzione corrispondente; in caso contrario, rimane sempre l'approccio opposto: provare funzioni a caso per vedere a quale tipo di grafico danno origine. Sperimentate, e non mancate di farci pervenire i vostri migliori risultati!

□

3D Professional

Un nuovo e potente programma di modellazione e rappresentazione tridimensionale, particolarmente dotato dal punto di vista dell'interfaccia grafica, e di uso molto intuitivo.

di ANTONIO DE LORENZO
Prima parte

Mentre da un lato si assiste ad un crescente livello di sviluppo e sofisticazione nelle tecniche di rappresentazione e calcolo di grafica tridimensionale, dall'altro vanno contemporaneamente affermandosi lo studio e la realizzazione di interfacce utenti sempre più semplici ed intuitive. A cosa servono infatti programmi dotati di enormi potenzialità se queste ultime non vengono poi sviluppate ed utilizzate appieno?

Con «**3D Professional**» (chiamato «3DPro» d'ora in poi) la Progressive Peripherals & Software ha compiuto uno sforzo notevole per offrire un ottimo programma di grafica dotato di un'interfaccia utente curatissima e funzionale.

L'utente mantiene sotto controllo tutte le funzioni del programma: la loro attivazione risulta veloce e la rappresentazione estremamente curata. Coloro che si sono sempre scontrati con le difficoltà proprie dei programmi di grafica 3D, potranno finalmente raggiungere risultati apprezzabili, così come i più smaliziati potranno ulteriormente affinare ed estendere le proprie esperienze grafiche.

Dato il numero di funzioni e strumenti messi a disposizione dal package (solo il programma principale occupa oltre 600 Kb), divideremo questo tutorial in due puntate: seguiteci con attenzione... il programma ne vale davvero la pena!!!

IL MATERIALE

«3DPro» si presenta in una veste davvero ricca: in un elegantissimo contenitore verde smeraldo sono contenuti ben sei dischi, tre manuali ed una videocassetta tutorial della durata di due ore. Il pacchetto viene venduto in due versioni, **NTSC** (standard televisivo americano) e **PAL** (standard europeo, con maggior numero di righe verticali).

I sei dischi includono anche «**Animation Station**», un programma per la manipolazione di animazioni in formato Anim venduto anche separatamente, già analizzato nel fascicolo 19 di Amigabyte.

Gli altri quattro dischi contengono il programma principale (presente in versione normale per il processore 68000 ed in versione Floating Point per schede accelleratrici con processore matematico e modelli Amiga della serie 2500 e 3000); completano il contenuto una nutrita serie di esempi e di dimostrazioni dei risultati raggiungibili con il pacchetto.

IL PRIMO IMPATTO

Caricato il WorkBench, basta inserire nel drive il dischetto contenente il programma principale; due click con il mouse, una manciata di secondi, e si è pronti per iniziare. Per chi possiede un hard disk, l'installazione del programma viene effettuata automaticamente (è sufficiente inserire i disk richiesti durante l'install) da un apposito script, ma al termine bisogna aggiungere due Assign nella startup-sequence. Appena eseguito, il programma rivelerà lo standard televisivo in uso, il processore ed eventuali coprocessori, oltre che data ed ora. Immediatamente apparirà una tavola di allocazione (Fig. 1) per specificare la scena che andiamo a costruire. Se si dispone soltanto di 1 Mb di memoria, i valori di default presenti sono troppo elevati ed

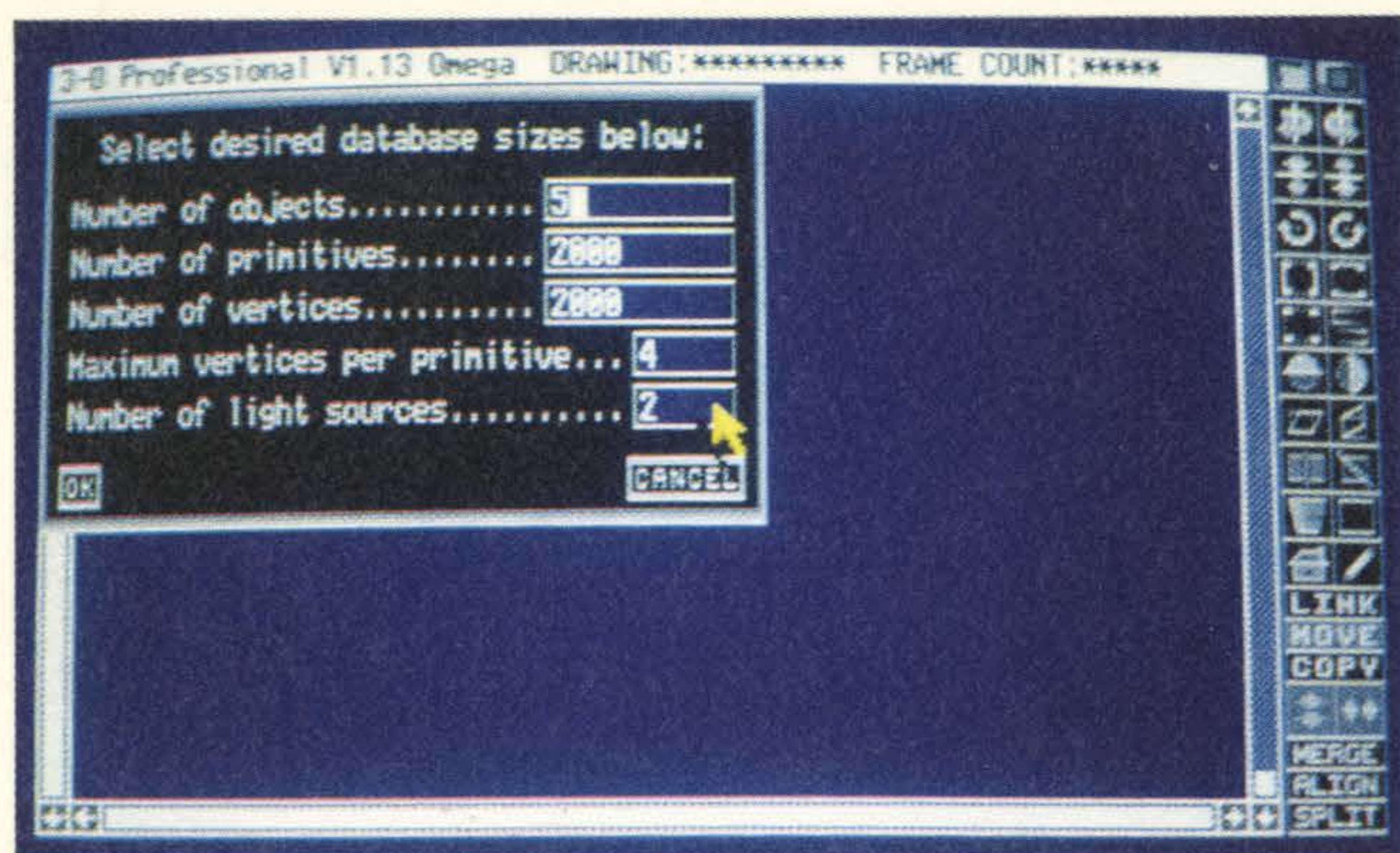



Fig. 1. L'inserimento dei valori nel Database di allocazione al lancio del programma.

occorrerà ridimensionarli. Questa fase è molto delicata ed importante per la buona riuscita del lavoro: è necessario quindi prestare particolare attenzione ai valori da inserire, poiché condizioneranno le scelte future. Per maggior chiarezza esemplifichiamo il riempimento della tabella (database) con un esempio per la costruzione di un cubo. **Number of object** seleziona il numero di oggetti da inserire, nel nostro caso 1; **Number of primitives** non specifica, come si potrebbe pensare, il numero di primitive che compongono l'oggetto, bensì le facce dell'oggetto stesso, nel caso del nostro cubo, quindi, 6; **Number of vertices** è il numero di vertici (8) mentre **Maximum vertices per primitive** il numero massimo di vertici per faccia (4); **Number of light sources** è il numero di sorgenti luminose (a meno che non ricerchiamo particolari effetti, una singola fonte di luce risulta sufficiente).

Questa dichiarazione d'intenti previene eventuali visite del Guru per mancanza di memoria durante la fase di costruzione e di rendering della scena (sempre conoscere prima i propri limiti!) ma rende praticamente impossibile la modifica successiva di una stessa scena con un numero di oggetti, di vertici e di sorgenti luminose maggiore di quello dichiarato inizialmente. Se si dispone di oltre 1 Mbyte di memoria è buona regola inserire valori superiori alle intenzioni iniziali, per consentire una certa elasticità nella costruzione della scena e cautelarsi per eventuali modifiche successive. Se invece la memoria disponibile del sistema ammonta ad un solo Mbyte, allora per risparmiarne il più possibile è necessario attenersi a valori coincidenti o per lo meno molto vicini a quelli previsti.

Inseriti questi valori, il programma comunicherà la fine delle operazioni di allocazione e sarà pronto a ricevere i nostri ordini. Per evitare di specificare di volta in volta tali valori insieme ad altri parametri presenti nel programma, si può decidere di salvare su disco quanto appena settato.

Fig. 2. L'editor principale di «3DPro». La visualizzazione opzionale degli assi e dei confini rende al meglio la rappresentazione tridimensionale della scena.

I MENU PROJECT E FILE

Inizializzato il programma, lo schermo risulta suddiviso in uno spazio che ne occupa la maggior parte (l'editor di «3DPro» è l'accesso al suo universo) contornato dai gadget di spostamento e di ingrandimento su tre lati, e in un Toolbox laterale (Fig. 2). I confini del nostro mondo sono segnati da un quadrato costituito da linee tratteggiate. In alto ritroviamo, come di consueto, la **Title Bar** con ben dieci menu stracolmi di opzioni e sotto-opzioni che ora esamineremo.

Il primo menu che incontriamo è **PROJECT**, il quale contiene opzioni di uso generale:

– **ABOUT** fornisce notizie sulla versione utilizzata.

– **CREDITS** è un'originale idea per presentare ringraziamenti e riconoscimenti attraverso un'immagine grande quattro volte lo schermo e nella quale ci si muove spostando il mouse in tutte le direzioni. Ogni gruppo o persona che ha contribuito alla realizzazione di «3DPro» ha lasciato il suo segno con frasi o disegnini, dando libero sfogo alla propria fantasia: l'insieme ricorda le scritte che si appongono su un'ingessatura! Si esce premendo la barra spaziatrice.

– **NEW** inizia un nuovo lavoro senza dover ricompilare la tabella di allocazione.

– **ALLOCATE** ridistribuisce la memoria ripresentando la tabella di allocazione iniziale. È come se si sospendesse l'esecuzione (quit) e poi si ricaricasse il programma in quanto parametri interni, valori ed attributi vengono risettati come da default.

– **MEMORY** fornisce l'ammontare di Chip e Fast Ram.

– **WORKBENCH** permette di inattivare il WorkBench consentendo una maggiore disponibilità di memoria libera.

– **SPAWN** impartisce comandi da Shell.

– **COLLECT** «raccoglie» eventuali residui di memoria.

– **ICONIFY** «iconizza» il programma e ritorna al WorkBench senza sospenderne l'esecuzione. Per tornare allo schermo di lavoro, clickare sul gadget di chiusura della finestra con la dicitura «3D Professional».

– **QUIT** termina l'esecuzione del programma dopo aver chiesto conferma.

Secondo menu sotto esame: si tratta di **FILE**. Esso raccoglie per la maggior parte le opzioni tipiche di Input/Output, cioè di caricamento e salvataggio dati:

– **LOAD OBJECT** permette di caricare oggetti in una notevole quantità di formati: oltre che, naturalmente, il formato proprio, riconosce il formato GEO (Videoscape 3D), **Sculpt 4D**, Forms in Flight, 3D Demon, Turbo Silver, Autocad .DXF (IBM e compatibili), Cad 3D (Atari ST). La gran quantità di formati riconosciuti consente di disporre di un numero estesissimo di oggetti già pronti, senza necessità di utilizzare programmi di conversione. Se l'oggetto prescelto eccede i valori impostati in fase iniziale, tali valori andranno aumentati (memoria permettendo) o il programma non

potrà procedere al caricamento. Inoltre, una volta riconosciuto il tipo di dati, «3DPro» chiede se si desidera raddoppiare la definizione. Anche questa scelta sarà condizionata dai valori base impostati e quindi, in ultima analisi, dalla quantità di memoria presente sulla macchina. Si tenga presente, infine, che il programma non converte anche gli attributi propri dell'oggetto (materiale, proprietà ottiche di trasparenza, lucentezza, etc.) limitandosi alla sua forma.

– **SAVE OBJECT** salva l'oggetto selezionato tramite la funzione **Selector** (presente nel **MENU EDIT**). Occorre specificare nel request il nome ed il percorso (path) del file, oltre che il formato desiderato (3D Pro o Videoscape).

– **LOAD SCENE** carica un'intera scena in formato 3D Pro.

– **SAVE SCENE** salva una scena in formato 3D Pro.

– **SHOW ILBM** mostra una schermata in formato IFF.

– **SAVE ILBM** salva la scena in formato IFF.

– **SHOW XS3D** mostra l'immagine sdoppiata presa da due punti di vista leggermente differenti. L'utilizzo degli speciali occhiali (gli «**X-Specs**» della Haitex) consentirà una visione stereoscopica, cioè tridimensionale (l'immagine appare in rilievo nonostante venga visualizzata su di una superficie piana).

– **SHOW RGB** mostra le immagini in **formato RGB**.

– **RUN SCRIPT** consente il caricamento di file script elaborati sia in linguaggio proprio che in **Arexx**. Usando i due modi appropriatamente, possiamo sfruttare appieno il programma specialmente per ciò che riguarda la costruzione e il calcolo delle animazioni, automatizzando i processi più tediosi (ad esempio il calcolo dei fotogrammi intermedi detto **key by frame**). Per elaborare i file script è necessario un qualsiasi editor di testi (per esempio «**Cygnus**», o «**ED**» del CLI), o un word-processor che salvi in formato ASCII (come «**C1-Text**», per esempio).

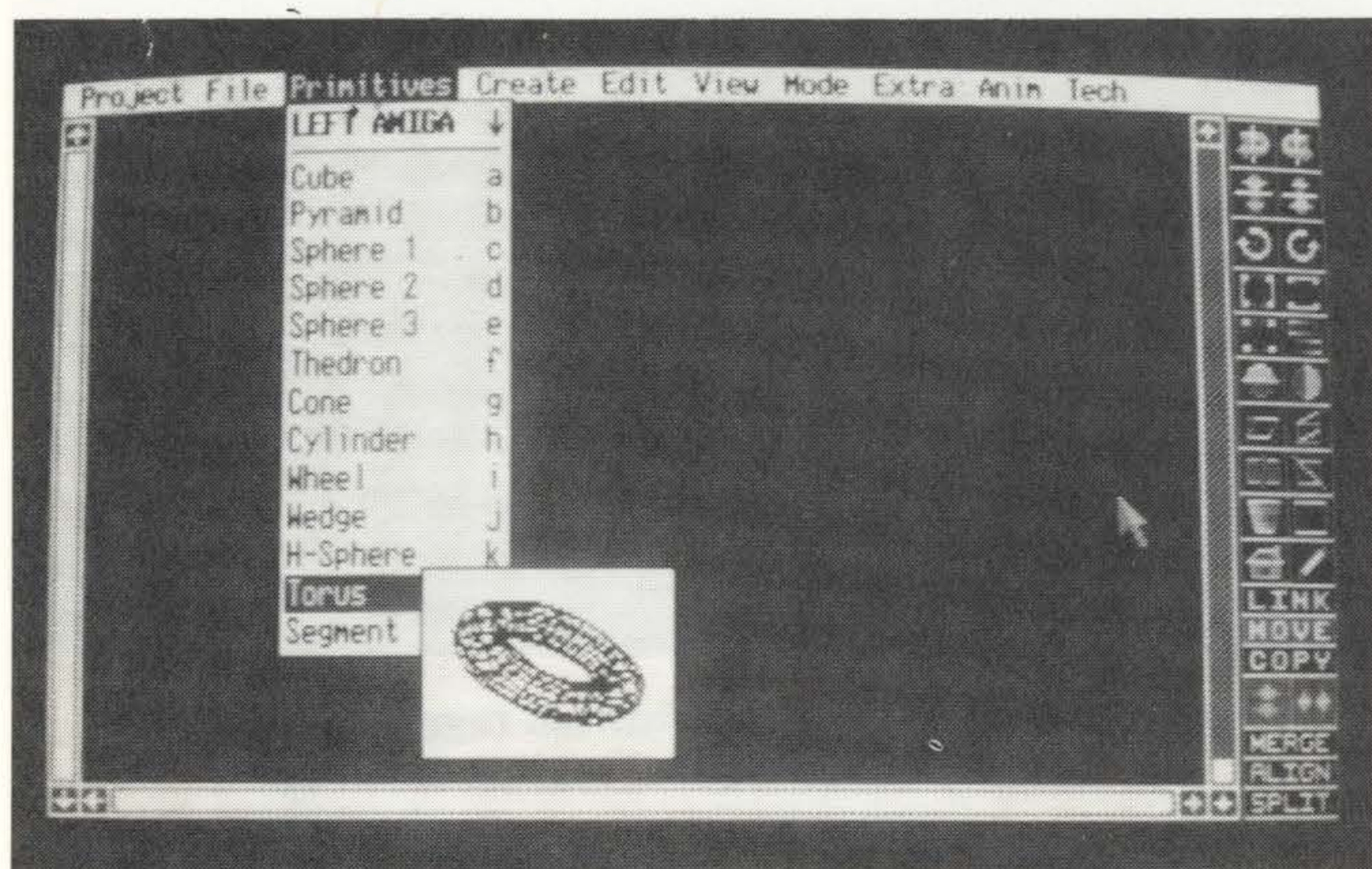



Fig. 3. Numerosi solidi geometrici sono accessibili dal menu Primitives.

LE PRIMITIVE

Come spesso vari pacchetti grafici di modellazione solida, anche «3DPro» fa uso esteso di primitive, vale a dire di figure tridimensionali già pronte e subito utilizzabili nella scena. Mediante il **MENU PRIMITIVES** (Fig. 3) si possono aggiungere ben 13 tipi diversi di primitive (cubo, piramide, tre tipi di sfere di crescente complessità, tetraedro, cono, cilindro, ruota, cuneo, semisfera, toro e segmento). I solidi geometrici qui indicati come primitive non sono gli stessi identificati in apertura nella dichiarazione del file. In quel caso, infatti, ricordiamo che per primitive intendevamo il numero di facce che compongono un solido.

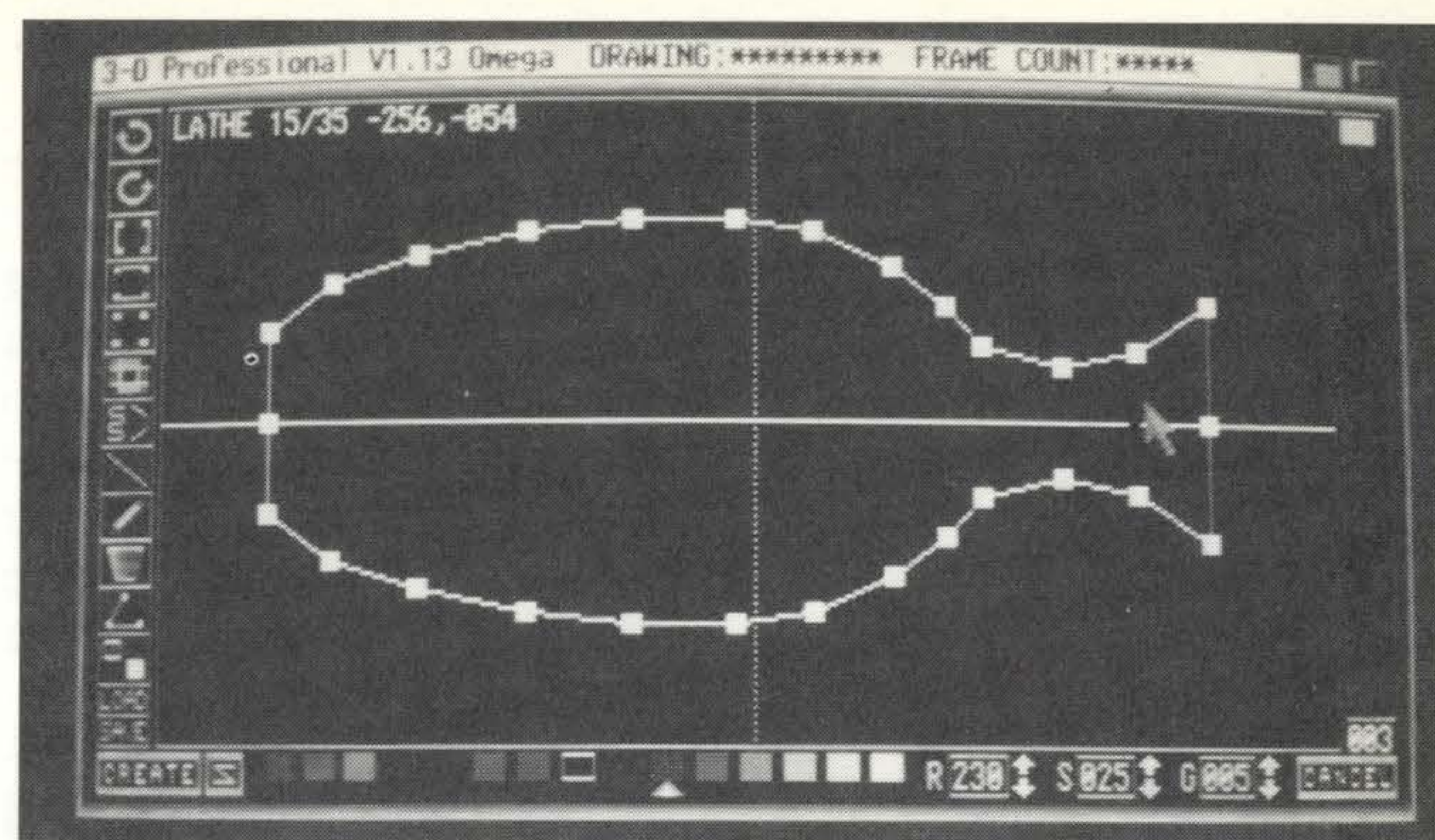



Fig. 4. L'editor Lathe è simile nella sua composizione agli editor Profile e Conic.

Uno degli aspetti finora trascurato dai programmi di modellazione 3D è stata la costruzione e rappresentazione degli scenari. Spesso si notano oggetti di per sé molto belli ma isolati, non inseriti in un preciso contesto. Da questo punto di vista «3DPro» offre alcuni dei più potenti e versatili editor sul mercato. Il menu contenente otto tool per la generazione di oggetti e ambienti è **CREATE**.

CREARE UN MONDO

Tranne alcune differenze che tra breve enumereremo, l'editor dei primi tre tool (**Lathe, Profile, Conic**) è molto simile (in Fig. 4, l'editor Lathe). Regola comune a tutti e tre i tool è che le varie forme *devono sempre essere rappresentate da poligoni chiusi* utilizzando l'apposito gadget presente nel toolbox sinistro (identificabile perché rappresentato con un triangolo inizialmente aperto, che si chiude dopo la selezione). Inoltre ogni punto (evidenziato da un quadratino e da un puntino centrale) può essere agevolmente spostato per trascinamento dopo aver clickato all'interno (diverrà pieno).

Iniziamo con **Lathe**. In inglese questo termine significa tornio e proprio questa è la funzione cui assolve il sotto-editor; per mezzo degli strumenti presenti si può infatti costruire un solido di rotazione a

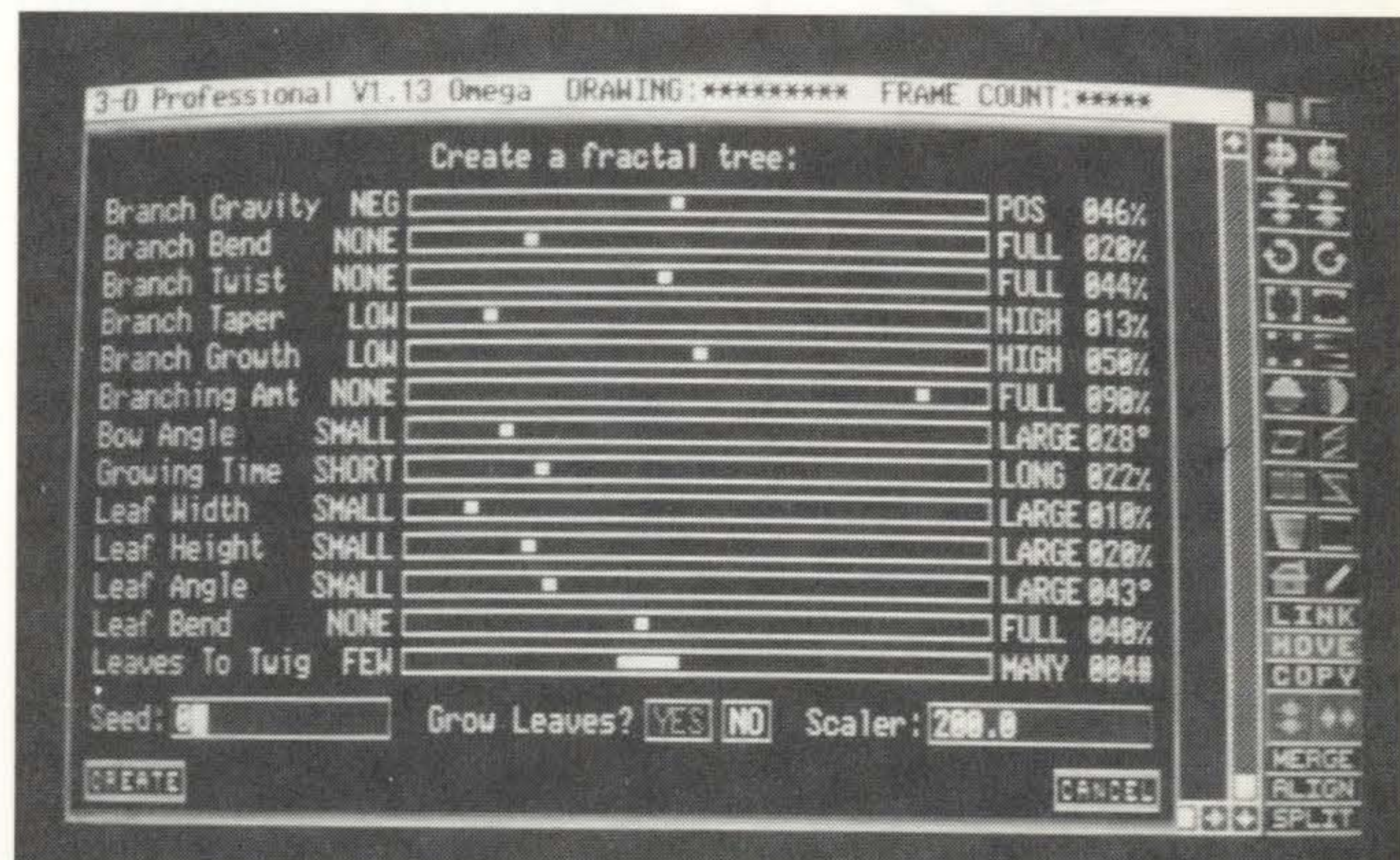



Fig. 5. I numerosi parametri per la definizione di un albero in geometria frattale.
I singoli valori vanno inseriti muovendo alcuni cursori.

partire dal disegno del profilo.

Sulla sinistra è presente una serie di comandi il cui significato è simile a quello delle altrettanto simili icone del Toolbox laterale dell'editor principale. Esse sono identiche in tutti e tre i tool in questione.

Il disegno viene effettuato per spostamenti con il mouse e tramite pressioni del tasto sinistro, il quale lascia sulla sua traiettoria una serie di punti (racchiusi da un quadrato) che rappresentano i vertici di una singola sezione. Affinché si possano scegliere le operazioni associate alle icone, è necessario selezionare i punti sui quali si vogliono applicare le singole operazioni. La selezione di tutti i punti (rappresentata dal riempimento dei quadratini) così come la loro deselezione, è compito dell'icona contenente due quadratini (uno vuoto ed uno pieno), mentre si possono scegliere i singoli punti da selezionare clickandoci sopra con il mouse. Si possono ruotare i punti selezionati in senso orario ed antiorario (prime due icone in alto); riscalare la figura (tre icone successive), mentre l'icona quadrettata successiva effettua simmetrie rispetto agli assi o all'origine (dipende dal punto dell'icona sulla quale si clicka). Il cestino cancella quanto tracciato mentre la selezione delle icone **Load** e **Save** carica o salva quanto disegnato. In basso sono presenti le opzioni di scelta dei colori. La scelta di **CREATE** provvede alla generazione.

Ciò che è necessario sapere per il disegno di forme tridimensionali è che è sufficiente recarsi con il mouse sulla finestra dell'editor con i quattro assi e realizzare un profilo nella metà soprastante, tenendo il tasto del mouse premuto e spostandosi nelle direzioni desiderate. Il programma provvederà al disegno dello stesso profilo nella parte sottostante. Lo slide trascinabile sulla destra indica il numero di sezioni che comporrà l'oggetto (valore riportato numericamente alla base dello stesso).

— **PROFILE** provvede all'estrusione di forme disegnate nell'editor (quello che in gergo viene definito 2D e mezzo). L'estrema destra di questo è occupato per intero da due colonne, una delle quali reca una freccia. Selezionandola si possono generare altre frecce che, spostate, rappresentano le singole sezioni laterali sia in numero (dato dal numero delle frecce meno una) sia in estensione (rilevabile e definibile come distanza tra due frecce continue).

Infine, l'ultimo tool definito **Conic** è preposto alla realizzazione di coni. Il centro degli assi rappresenta la *proiezione dell'altezza sulla base*; quindi, se si desidera che l'altezza cada all'interno della base, quest'ultima deve essere disegnata con il centro degli assi al suo interno. Il gadget laterale destro indica la dimensione dell'altezza del cono.

— **LAND** crea un paesaggio frattale con tanto di acqua, colline e montagne. Il metodo è il medesimo utilizzato dal programma «**Scene Generator**» ed i risultati sono altrettanto brillanti. Ogni componente del paesaggio è costruito tramite la suddivisione di triangoli. Pertanto, **Level** specifica il livello di dettaglio. Quanto maggiore sarà il valore inserito (1-5), tanto più saranno piccoli e numerosi i triangoli, tanto maggiori risulteranno la memoria occupata, il tempo di costruzione e quello di rappresentazione (oltre che quello di rendering). **Roughness** (0-255) rappresenta il livello di asperità presente nel paesaggio; **Seed** un numero a piacere, costituisce il seme utilizzato per la costruzione ed assicura l'unicità del paesaggio, mentre gli ultimi due valori rappresentano la scala di costruzione.

Fig. 6. Un bell'esempio dei risultati raggiungibili con la funzione Tree e con il texture Snowy.

— **TREE** è un altro potente comando che, sempre utilizzando tecniche frattali, costruisce alberi con risultati che, oltre che essere unici, sono veramente notevoli: qualcosa come 16 parametri diversi specificano la forma della pianta e le eventuali foglie associate! I singoli valori vanno inseriti muovendo alcuni cursori (Fig. 5). **Branch Gravity** è la forza di gravità applicata ai rami dell'albero. Tanto maggiore risulta il suo valore, tanto più saranno piegati verso il basso i rami. **Branch Bend** specifica la curvatura che i rami dovranno assumere, **Branch Twist** «l'attorcigliamento»

dei rami su loro stessi, **Branch Taper** l'assottigliamento alle estremità (quanto più si sceglie un valore alto, tanto più tozzi e corti risulteranno i rami). **Branch Growth** stabilisce le dimensioni della pianta (sta a voi creare una quercia o un bonsai!); **Branching Amt** decide il grado di ramificazione. **Bow angle** richiede un valore in gradi ed esprime l'espandersi dei rami nelle varie direzioni (180 gradi per tutte le direzioni). **Growing time** stabilisce la lunghezza dei rami e del tronco.

L'insieme di questi valori definisce la forma del tronco e delle sue ramificazioni. Se intendiamo dotare di fogliame il nostro albero, sarà necessario selezionare **Yes** nella richiesta in basso (**Grow Leaves**), ed inserire i seguenti valori: **Leaf Width** e **Leaf Height** specificano rispettivamente la larghezza e la lunghezza delle foglie mentre **Leaf Angle** e **Leaf Bend** l'angolo con il quale le foglie fuoriescono dai rami, e la loro curvatura. Infine, troviamo **Leaves to Twigh**: questo parametro consente di specificare il grado d'infoltimento della chioma. Più alti sono i valori, maggiore sarà la quantità di foglie associate per ramo.

NASCE LA PIANTA

Una volta introdotti i valori descritti (modificandoli oculatamente, è possibile ottenere praticamente ogni tipo di albero o di arbusto) inserite un numero a piacere a fianco alla dicitura **Seed** (una nota di colore: la parole seed specifica un numero casuale e significa in inglese seme, proprio il seme che farà generare la nostra pianta) e clickate su **Create**. Dopo un tempo variabile da pochi secondi a qualche minuto (dipende dalla complessità) una bella pianta farà mostra di sé nel centro dello schermo.

Sempre sotto lo stesso menu è presente un'altra potente opzione, **Text** che, come il nome lascia intendere, ci consente di inserire del testo all'interno delle nostre scene o animazioni (per esempio nella realizzazione di sigle). Una volta invocata Text, appare un request ricco di sotto opzioni: **Select font** consente di specificare il font, che va selezionato nell'apposita directory sotto forma del suo valore numerico (corpo del carattere), cioè la dimensione del font stesso; **Point size** riporta la grandezza di un pixel presente nel font originario come blocco, che sarà tanto più grande quanto maggiore è questo valore; **Justification left, center, right** posizionano quanto scritto con giustificazione a destra, al centro o a sinistra. In **Text** va inserito il testo che si vuole trasformare in oggetti solidi mentre **Style I, B, U** associa lo stile (corsivo, neretto e sottolineato, con possibilità di combinazioni di due o di tutti e tre gli stili); **Text direction**, infine, direziona il testo in otto direzioni nello spazio. È possibile importare tutti i font standard. In Fig. 7 visualizziamo i risultati di **Text**: l'unico appunto che si può muovere all'operato di questa funzione è l'eccessiva seghettatura dei caratteri.

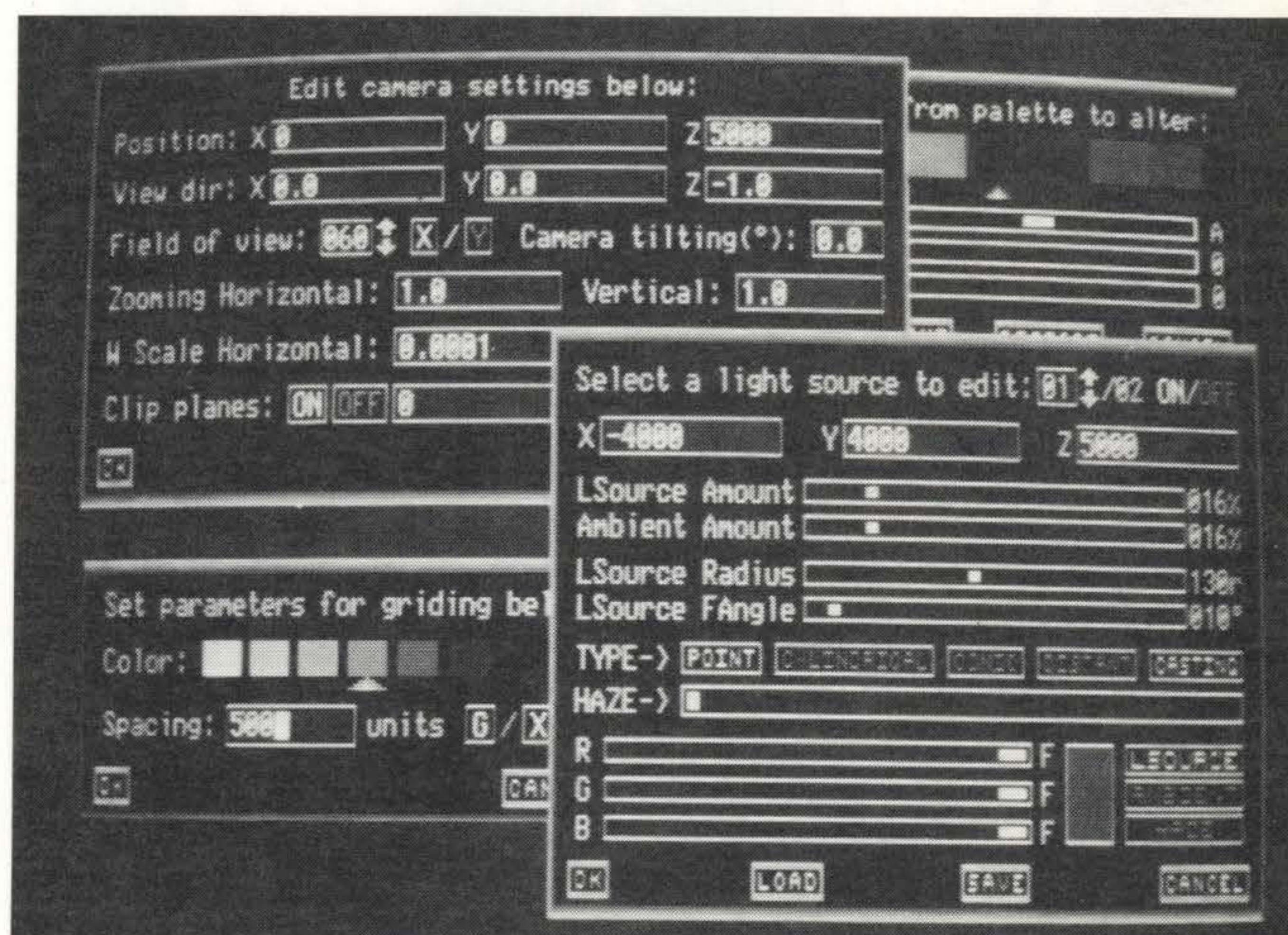
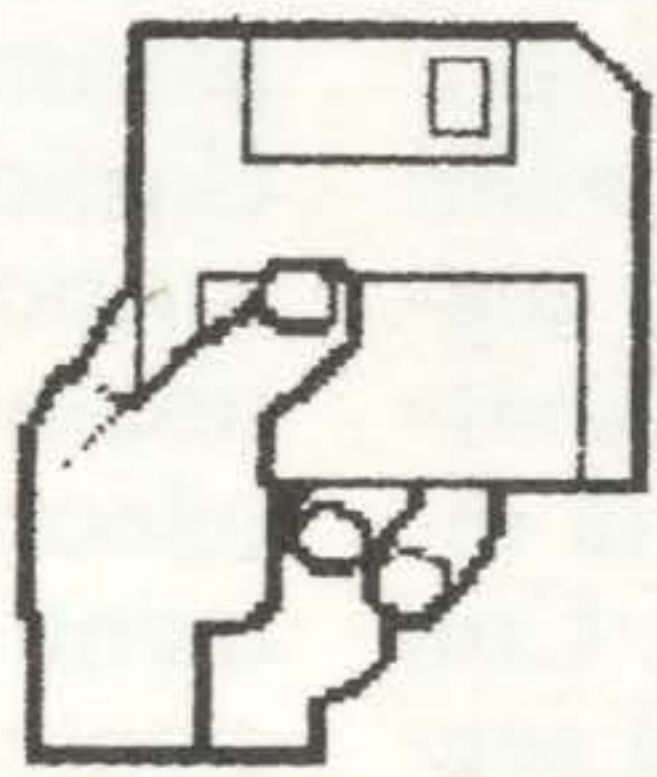
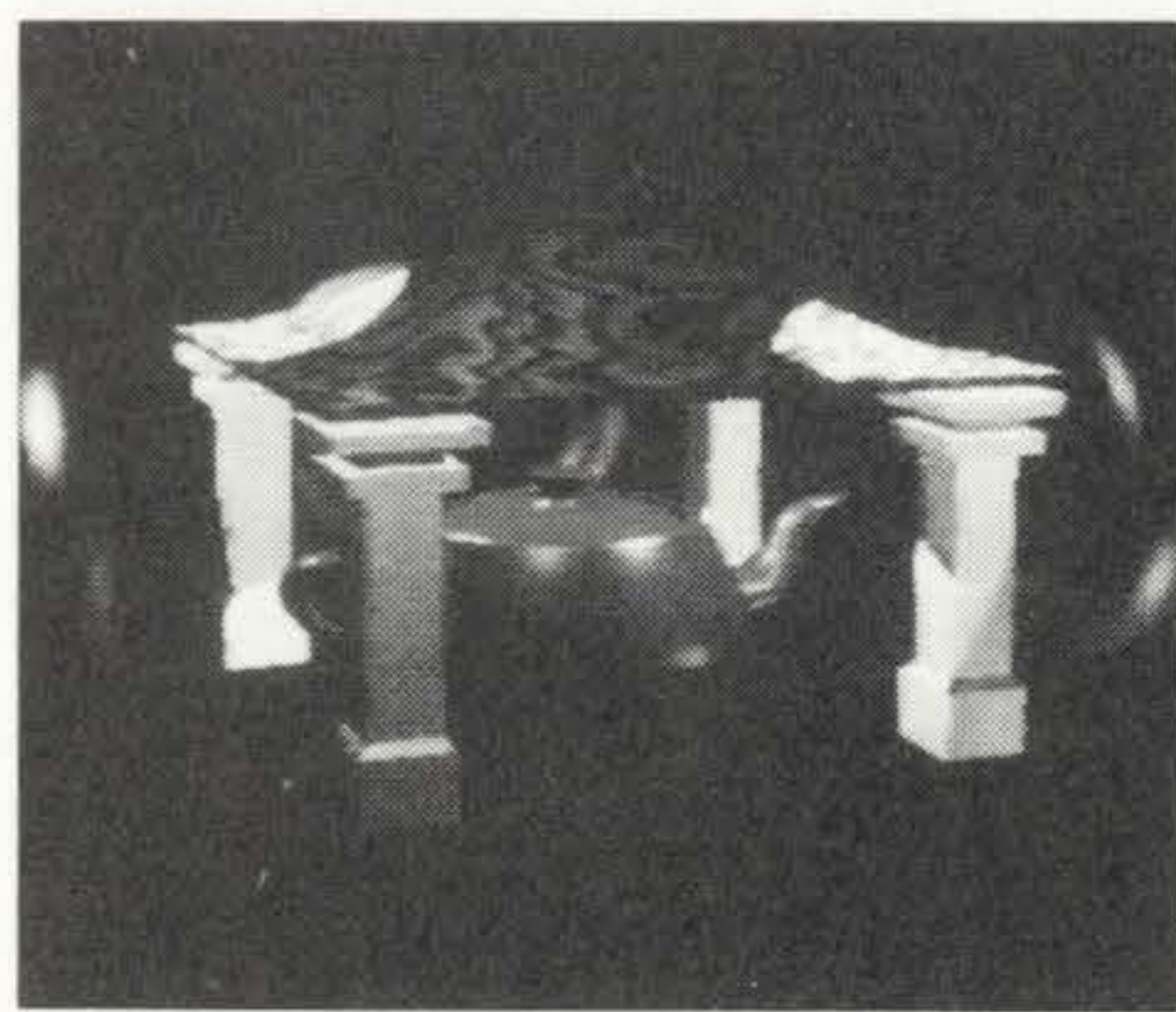
Continuando nell'esame, troviamo **ILBM to 3D**, in grado di convertire un brush IFF in un oggetto tridimensionale.

La creazione di un **GROUND**, cioè di un piano, non comporta particolari difficoltà. Si può scegliere tra **Solid, Wire** e **Checkered** cioè in un unico colore, a reticolo, oppure a scacchi. **Color 1** e **Color 2** definiscono i colori mentre i valori contenuti in **X** e in **Z** definiscono le dimensioni.

Successivamente troviamo il menu **EDIT**. Le opzioni presenti sono le più complesse e numerose di tutto il programma. Esse riguardano per lo più la definizione e l'elaborazione sia degli oggetti inseriti nella scena che degli strumenti associati (come ad esempio il punto di vista, le sorgenti luminose, i colori, etc.).

Fig. 7. L'opzione Text consente di trasformare velocemente qualsiasi font in un oggetto tridimensionale.

— **CAMERA** Molteplici sono le modifiche che si possono apportare alla telecamera (Fig. 8), che in «3DPro» è rappresentata da un quadratino affiancato da una «C» dalla quale si diparte un segmento. Il quadratino può essere spostato a piacimento nello schermo, così come il segmento che da esso si diparte e che rappresenta la direzione dell'obiettivo. Con il mouse ci si reca sul segmento e, mantenendo premuto il tasto destro, ci si sposta sullo schermo. Il segmento seguirà docilmente il mouse e si allungherà, permettendo quindi la sua congiunzione con il centro dell'oggetto da inquadrare (quadratino verde), o nella direzione che si ritiene più appropriata. Questo sistema risulta molto intelligente e intuitivo, e il puntamento della telecamera diviene quindi facile, preciso e immediato. Il menu EDIT permette anche di settare tutta una serie di parametri rendendone più elastico l'uso: **Position x, y, z**, riporta in coordinate cartesiane la posizione della telecamera. Per spostamenti di precisione è sufficiente inserire i valori negli appositi spazi. **View dir**,

Fig. 8. I requester Edit, Lighting, Palette e Griding. Si noti la quantità di parametri e di opzioni presenti.

sempre in coordinate cartesiane, descrive la direzione dell'obiettivo e **Field of view x, y** rappresenta il campo visivo della telecamera: con valori elevati si vedrà l'oggetto inquadrato allontanarsi dalla telecamera perché più grande è il campo, mentre con valori minori l'effetto è quello opposto. Dato il gran fluttuare di questi valori, per permettere una rapida introduzione del valore da associare, oltre che usare le due frecce poste vicino alle rispettive opzioni per aumentarlo o diminuirlo, un click del mouse all'interno

del riquadro consente d'introdurre immediatamente il valore. **Camera tilting** esprime in gradi l'inclinazione laterale della camera. **Zooming Horizzontal** e **Vertical** definiscono lo zoom nelle due direzioni.

— **LIGHTING** è l'opzione che attiene alle fonti di luce dell'universo «3DPro». Se ne possono inserire fino a 99 (Fig. 8). Nell'editor grafico le fonti di luce sono rappresentate con un quadratino ed un puntino al centro affiancato dalla lettera «L». Come nel caso della telecamera, anche le fonti di luce possono essere facilmente spostate recandovicisi sopra e tenendo premuto il tasto sinistro del mouse durante lo spostamento. Se dal **MENU VIEW**, che esamineremo subito dopo, viene selezionata l'opzione **Polygon Shade**, lo spostamento della fonte di luce produrrà sulla superficie dell'oggetto un'ombreggiatura molto utile per regolarsi circa la presenza o meno di lati in ombra. Invocando **Lighting** apparirà un apposito request. Sono numerosi, al solito, i parametri e le scelte operabili. La prima scelta riguarda la selezione della fonte di luce da editare. Come abbiamo visto in apertura di articolo, il numero delle sorgenti di luce deve essere dichiarato all'inizio. Nel caso di più di una fonte, soltanto una viene aggiunta nella scena; le restanti sono disattivate e solo intervenendo in **Select a light source to edit** si selezionano le rimanenti fonti per decidere se attivarle o meno. Subito sotto sono riportate le coordinate cartesiane delle sorgenti luminose, modificabili più velocemente posizionandole con il mouse. **LSource Amount** esprime l'intensità della sorgente luminosa.

Type descrive il tipo di sorgente luminosa, che può essere puntiforme (**Point**), cilindrica (**Cylindrical**), conica (**Conic**), lontana (**Distant**), e diffusa (**Casting**) senza visione della sorgente. Il cursore associato ad **Haze** quantifica la presenza e la quantità di foschia.

In basso troviamo sia la familiare rappresentazione per la selezione dei colori delle sorgenti luminose dell'ambiente e della foschia, che le opzioni per salvare i valori introdotti e ricaricarli in altre scene. La flessibilità raggiunta nella manipolazione delle sorgenti luminose è ragguardevole, e si possono creare con notevole facilità effetti luminosi tipo laser o fasci di luce.

La prima parte del tutorial termina qui. Nella seconda avremo modo di terminare la parte teorica, descriveremo i restanti menu ed opzioni, e passeremo alla pratica con la realizzazione di una scena completa. Vedremo inoltre come costruire, in due modi diversi, un'animazione. □

Fast Basic 2.0

Il basic gode ottima salute! Dopo «Amos», ecco a voi una nuova implementazione del linguaggio Basic che pone l'accento sulla velocità di esecuzione, pur restando facile da apprendere e da usare

di FABRIZIO LODI

Il linguaggio Basic, un tempo dato per spacciato, sta vivendo un periodo di rinnovata popolarità e vitalità grazie all'introduzione di nuove e più potenti implementazioni.

Nel 1990 hanno infatti visto la luce tre innovative versioni del Basic, caratterizzate soprattutto da prestazioni di tutto rispetto sotto il profilo della grafica e della velocità di esecuzione: l'inglese Mandarin ha introdotto l'interprete «**Amos**», erroneamente ritenuto da molti un mero creatore di videogiochi; l'australiana Mast ha presentato «**Blitz Basic**», un compilatore che implementa un subset ridotto di istruzioni Basic dedicato alla gestione di grafica e sonoro, del quale AmigaByte si occuperà presto; ed infine la statunitense DelphiNoetic Systems ha progettato, prima delle tre in ordine cronologico, lo spartano ma velocissimo compilatore «**Fast Basic**», del quale ci accingiamo ora descrivere le caratteristiche salienti.

IL COMPILATORE

«Fast Basic», più noto come «**F-Basic**», non è un interprete bensì un compilatore, memorizzato sul dischetto con il nome «**FB**». Questo significa che, contrariamente a quanto avviene con «**AmigaBasic**», «**Amos**» o «**GfaBasic**», non è possibile digitare direttamente comandi o listati Basic ed eseguirli con il comando Run, ma occorre ogni volta compilarli, dopo essersi accertati che non contengano errori.

I listati dei programmi in «F-Basic» possono essere scritti mediante qualsiasi editor di testi: l'importante è che vengano salvati in formato Ascii. Inutile dire che, purtroppo, «F-Basic» non è compatibile con «AmigaBasic», pertanto non può essere usato per compilare listati scritti con l'interprete MicroSoft, a meno di convertirli manualmente.

Insieme al compilatore si trovano inoltre tre librerie, denominate «**FastLib**», «**FastSysLib**» e «**FastLinkLib**».

Le prime due sono librerie di *runtime*: ovvero la loro presenza è indispensabile al funzionamento di qualsiasi programma compilato con F-Basic. Possono trovarsi nella stessa directory del programma (o dei programmi) da eseguire, oppure nella **directory «S:»** del disco di sistema. La libreria «FastLib» è indispensabile a qualsiasi programma; la «FastSysLib» invece è necessaria solo se il programma deve accedere alle librerie di sistema di Amiga. Il fatto di usare librerie condivise rappresenta un vantaggio in termini di occupazione di spazio su disco: con questo sistema è necessaria la presenza di una sola copia della «FastLib» (ed eventualmente della «FastSysLib»), anche se su di un disco sono presenti più programmi F-Basic in directory differenti.

Chi volesse invece compilare un programma in modo

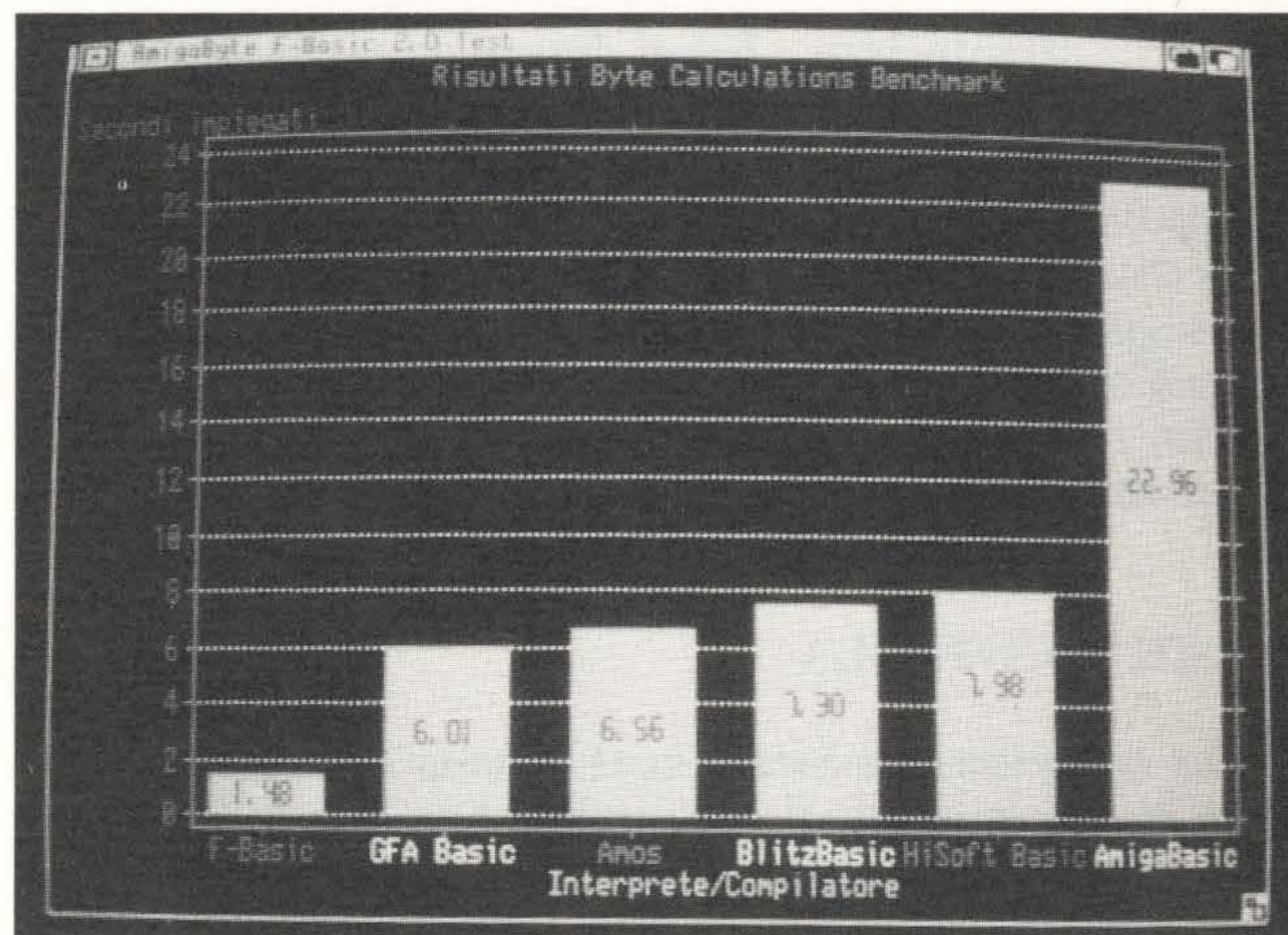



Tabella comparativa dei risultati del benchmark di Byte eseguito con altri Basic. «F-Basic» è di gran lunga il più veloce e preciso nei calcoli.

da renderlo completamente indipendente ed eseguibile senza bisogno delle librerie esterne, può ricorrere all'uso del programma «**Link**» fornito con il compilatore, che sfrutta la terza libreria presente sul disco (la «FastLinkLib»). Questa utility effettua il linking (ovvero l'unione fisica) di un file eseguibile con le librerie necessarie al suo funzionamento, generando un programma autonomo che non richiederà la presenza della «FastLib» per essere lanciato. L'evidente svantaggio consiste nel fatto che, ovviamente, i file trattati con «Link» saranno considerevolmente più voluminosi di prima.

Il pacchetto «Fast Basic» è completato da un dischetto

```
PROGRAMMA <nome>
    <parte dichiarativa>
    <parte eseguibile>
    STOP
    {<etichetta 1>}
    <subroutine 1>
    .
    .
    .
    .
    {<etichetta N>}
    <subroutine N>
END
```

Fig. 1: la struttura tipica di un programma «F-Basic» contenente delle subroutine.

con circa ottantacinque listati dimostrativi, da una directory denominata «**FInclude**» contenente alcuni file di definizioni necessari per lo sfruttamento di alcune risorse del sistema operativo di Amiga, e da due ulteriori file indispensabili: il primo, «**FastError**», viene usato da «Fb» per segnalare gli errori in fase di compilazione; l'altro, denominato «**Sldb**», è il *Source Level Debugger* del pacchetto, e concorre all'individuazione ed all'eliminazione di eventuali errori all'interno dei programmi.

COME SI COMPILA

Per compilare un listato in «F-Basic» occorre utilizzare il Cli, richiamando il «Compiler» con questa sintassi:
FB nomefile [destinazione] OPT - opzioni
dove *nomefile* è ovviamente il nome del listato da compilare, eventualmente comprensivo di directory e nome del drive; *destinazione* specifica facoltativamente il nome da assegnare al file eseguibile generato da «FB»; *opzioni* indica uno o più parametri del compilatore.

Le direttive che possono essere impartite al compilatore sulla linea di comando sono le seguenti:
−C <byte>: cambia la lunghezza massima in byte del buffer usato per la concatenazione di stringhe. La sua ampiezza di default è di 1000 byte.
−G: specificando questa opzione il file verrà compilato in memoria invece che sul disco e verrà mandato immediatamente in esecuzione. Ovviamente, alla fine dell'esecuzione il file compilato andrà perso, quindi questa opzione si usa prevalentemente durante il debugging dei programmi.
−O <byte>: cambia la lunghezza massima in byte di un file oggetto. Il valore di default è di 50000 byte (50K).
−P: durante la compilazione, verrà visualizzata sullo schermo la linea del listato in fase di elaborazione. Ovviamente questa opzione allunga notevolmente i tempi richiesti per la compilazione.
−S <byte>: cambia le dimensioni del buffer dove vengono conservate le variabili contenute nella riga del listato in fase di compilazione. Il valore di default è 1000 byte.
−D: genera un file necessario al programma «SLDB», usato per il debugging.

I parametri devono essere scritti tutti di seguito. Ecco un esempio:
Fb df0:prova.o df1:prova.run OPT-C4000-D-P
Digitando questa linea si impartirà al compilatore l'ordine di compilare il file di nome «prova.o» presente nel drive df0: e di generare sul drive df1: un file eseguibile chiamato «prova.run», cambiando la lunghezza massima del buffer per la concatenazione di stringhe in 4000 byte. Le altre opzioni fanno inoltre in modo che venga generato il file di informazioni necessario al debugger «SLDB» e che vengano stampate sullo schermo le linee mentre vengono compilate.

DIRETTIVE AL COMPILATORE

Come molti compilatori di altri linguaggi, anche l'«F-Basic Compiler» accetta comandi particolari (definiti in genere «**metacomandi**») all'interno dei listati per attivare alcune particolari funzioni. Questi comandi si differenziano dai comandi Basic grazie al fatto che vengono interpretati ed eseguiti da «F-Basic» solo in fase di compilazione, ed influenzano alcune caratteristiche del linguaggio.

Vediamoli in dettaglio:
&DELIMITER <carattere>: serve per cambiare i delimitatori delle stringhe dai doppi apici standard al carattere passato al comando.
&EDITOR <comando>: specificando questa direttiva,

LE STRUTTURE DI CONTROLLO

Ecco un elenco delle strutture di controllo del flusso dei programmi in «F-Basic», con un breve commento ove l'uso differisca da quello degli altri linguaggi:
IF ... THEN: Esistono quattro modi di usare la struttura IF ... THEN. L'uso è pressochè identico a quello di tutti gli altri linguaggi; ricordiamo solo che non esiste nessun limite al numero delle istruzioni in <istruzioni> nè al numero di ELSE IF, tranne ovviamente la memoria disponibile:
IF <condizione> THEN <istruzione>
IF <condizione> THEN <istruzioni> END IF
IF <condizione> THEN <istruzioni> ELSE <istruzioni> END IF
IF <condizione> THEN <istruzioni> ELSEIF <condizione> THEN <istruzioni> ELSE <istruzioni> END IF
WHEN ... ENDCASE: Questa è una versione del più comune comando CASE. Il <selettore> può essere una qualunque espressione, mentre <val1> ... <valN> sono i valori dell'espressione che si vogliono controllare. Il programma eseguirà le istruzioni corrispondenti al valore <valN> uguale a quello dell'espressione definita in <selettore>; altrimenti verranno eseguite le istru-

quando il compilatore incontrerà un errore verrà eseguito da CLI il comando indicato, che potrà essere usato per richiamare un editor.
&EVENT_CHECK n: con questo metacomando si può abilitare (n=1) o disabilitare (n=0) il controllo di eventi che, in ogni caso, verrà ripreso quando verrà incontrata nel listato l'istruzione **ON EVENT**. Questo metacomando può essere utile per risparmiare tempo in parti di programma nelle quali un evento già attivato non serve più, visto che il controllo di eventi porta via molto tempo al 68000.
&FILESEP <carattere>: normalmente gli elementi scritti da un programma in un file di output vengono separati da una virgola. Tramite questo metacomando, invece che la virgola si può definire un altro carattere come separatore.
&QUICK n: con n=1 il compilatore, eseguendo operazioni su numeri interi, darà per scontato che i numeri siano inferiori a 32767, guadagnando in velocità; con n=0 viene riabilitato il normale controllo sulle dimensioni degli interi.
&SEPARATOR <carattere> / <#codice ascii>: cambia il separatore delle variabili di un comando di input. Può essere specificato direttamente il nuovo carattere da adottare come separatore, oppure il suo codice Ascii.
&IGNORE n: se n=1, il compilatore tratterà tutte le linee seguenti come commenti fino al prossimo &IGNORE 0, ovvero non le compilerà.
&SYSLIB n: questa direttiva segnala l'uso (n=1) o meno (n=0) delle librerie di sistema. In caso positivo, è indispensabile la presenza della libreria «FastSysLib».
&MENUWIDTH <pixel>: determina la larghezza in pixel delle barre dei vari menu a discesa eventualmente presenti nel programma, attivabili con il tasto destro del mouse.
&ANCHOR n: ponendo n=0 (&ANCHOR 0) si dirà al compilatore di usare, nei comandi relativi al pattern matching, il cosidetto *unanchored mode*; mentre con n=1 si riattiverà il modo standard, ovvero il modo *anchored*.

zioni specificate dopo l'istruzione OTHERWISE, se presente.
WHEN <selettore> IS
  [<val1>] <istruzioni_ 1>
  :   :
  :   :
ENDCASE
DO... WHILE: Esegue le istruzioni ed esce quando la condizione specificata non è più vera; se si arriva a questa istruzione quando la condizione è già falsa, il ciclo WHILE verrà saltato direttamente.
WHILE <condizione> DO
    <istruzioni>
ENDWHILE
REPEAT...UNTIL: A differenza di WHILE, la condizione specificata è una condizione di ingresso; cioè le istruzioni contenute nel ciclo verranno eseguite fintanto che la condizione è falsa: si uscirà dal loop quando la condizione diventerà vera. Poiché la condizione viene testata alla fine del ciclo, le istruzioni verranno eseguite sicuramente almeno una volta.
REPEAT
    <istruzioni>
UNTIL <condizione>
I cicli ON ... GOTO e FOR ... NEXT hanno in «F-Basic» la stessa struttura comune a tutti gli altri Basic.

```
CED File: Esempio.bas
PROGRAM Esempio
        GLOBAL
        INTEGER I
        LOCAL
        INTEGER X
        SUBPROGRAM
                INTEGER Funzione
                SUBROUTINE Ciao
        DATA (I,5,X,2)
        CALL Ciao(I)
        PRINT Funzione(X)
END

SUBROUTINE Ciao
        PARAMETER
        INTEGER Y
        LOCAL
        STRING T*4
        DATA (T,"Ciao")
        PRINT T,Y
END

FUNCTION Funzione
        PARAMETER
        INTEGER Z
        LOCAL
        INTEGER W
        W = 5
        Funzione = W + Z + I
END
```

Fig. 2: esempio di utilizzo delle direttive Global e Local nella parte principale di un programma, in una subroutine ed in una funzione.

STRUTTURA DEI PROGRAMMI

In «F-Basic» un programma presenta normalmente la seguente struttura:
PROGRAM <nome>
  <parte dichiarativa>
  <parte eseguibile>
END
Nella parte dichiarativa devono essere appunto dichiarati il tipo ed i valori iniziali delle variabili e delle costanti, oltre che i nomi delle funzioni e quelli delle subroutine usate nel programma. La dichiarazione si dividerà in tre parti, precedute da altrettante parole chiave:
GLOBAL: precede la dichiarazione delle variabili globali, cioè accessibili da qualsiasi parte del programma.
LOCAL: precede la dichiarazione delle variabili locali, ovvero quelle «visibili» solo dal programma principale e dalle routine in esso contenute.
SUBPROGRAM: precede la dichiarazione dei nomi delle subroutine e quelli delle funzioni esterne al programma principale, in questo formato. Le prime andranno dichiarate specificandone il nome preceduto dall'istruzione **SUBROUTINE**, mentre le funzioni devono essere dichiarate solo indicandone il tipo. Ciò è necessario perché in «F-Basic» le funzioni sono considerate alla stregua di variabili: le funzioni devono essere richiamate sempre uguagliandole a delle variabili (esempio, X=FUNZIONE(Y)) o usandole in strutture di controllo o in istruzioni di stampa, poiché restituiscono un valore; per assegnare ad una funzione questo valore bisogna perciò semplice-

Comando	Descrizione
CURS_DN(N)	Muove in basso il cursore N volte
CURS_UP(N)	Muove in alto il cursore N volte
CURS_LE(N)	Muove a sinistra il cursore N volte
CURS_RI(N)	Muove a destra il cursore N volte
CURS_LOC(X,Y)	Muove il cursore alla riga X, colonna Y
CURS_PO()	Ritorna la posizione del cursore
CURS_INV	Il cursore diventa invisibile
CURS_VIS	Il cursore torna visibile
ERA_ALL	Cancella lo schermo dal cursore alla fine del testo
ERA_EOL	Cancella il testo dal cursore alla fine della linea
ERA_LIN	Cancella la linea su cui si trova il cursore
INS_LIN	Inserisce una nuova linea
SCR_BEEP	Fa un flash sullo schermo
SCR_CLER	Cancella lo schermo e porta il cursore in alto a sinistra
SCR_FORE N	Cambia a N il colore di primo piano
SCR_BACK N	Cambia a N il colore di sfondo
SCR_STYL N	Cambia lo stile di stampa: N=0 Normale; N=1 Grassetto; N=2 Corsivo; N=3 Sottolineato; N=4 Negativo

Fig. 3: i comandi di «F-Basic» relativi al controllo del cursore e dell'output di testo sullo schermo.

mente usare il nome della funzione stessa come se fosse (e lo è) una variabile.

La parte dichiarativa contiene inoltre la definizione delle istruzioni **DATA**. Quest'ultimo comando ha una sintassi leggermente diversa da quella solitamente adottata nei normali interpreti Basic:

DATA (var1, var2, ..., valN) oppure DATA (array, val1, ..., valN)

In «F-Basic» l'istruzione DATA serve ad assegnare valori a delle variabili ma, diversamente dal solito, senza dover leggere i dati con un comando **READ**. Naturalmente è possibile usare anche una combinazione dei due modi visti sopra.

Le variabili si dichiarano invece in questo modo:

Tipo_variabile <nome1, ..., nomeN>

Al posto di *tipo_variabile* bisogna mettere un'istruzione che specifichi il genere di variabile definita. I valori possibili sono: **INTEGER, REAL, BYTE, WORD, TEXT** o **TEXT*lunghezza_stringa.**

Oltre che questi, esistono anche i puntatori (**PTR_TO**) e gli array. Per dichiarare questi ultimi bisogna seguire la sintassi usata per le variabili normali, aggiungendo le dimensioni desiderate secondo la forma (dim1, ..., dimN). Ad esempio:

INTEGER A(10,10)

equivale in Basic standard a:

DIM A(10,10)

con la differenza che, tramite la dichiarazione, in «F-Basic» si indica anche il tipo di elementi dell'array. Per assegnare valori agli array esistono due modi. Il primo consiste in un assegnamento:

<nome array>=<espressione> oppure <nome array1>=<nome array2>

Nel primo esempio viene assegnato il valore *espressione* a tutti gli elementi dell'array; nel secondo, l'array a destra del simbolo di uguaglianza viene copiato in quello a sinistra. Ovviamente i due array devono essere dello stesso tipo e devono avere le stesse dimensioni.

Il secondo metodo per assegnare valori ad un array sfrutta il comando DATA, la cui sintassi abbiamo visto precedentemente. Invece di elencare più volte un valore, si può anche indicare in «F-Basic» il numero di volte per cui deve essere inserito. Ad esempio:

INTEGER i(10)
DATA (i,10*5,5*5)

inserirà nei primi cinque elementi dell'array il valore 10 e nei cinque seguenti il valore 5. «F-Basic» permette anche di cambiare le dimensioni di un'array dopo la sua dichiarazione, nella parte eseguibile del programma, con il comando **SETARRAY**:

SETARRAY (nome_array, nuova_dimensione)

Per dichiarare le costanti, la procedura è simile a quella usata per le variabili, con la differenza che invece che il tipo di variabile si usa la parola chiave **CONSTANT** seguita dal valore della costante. Ad esempio:

CONSTANT j=3,i="prova"

Alla parte dichiarativa segue quella eseguibile, composta dai comandi veri e propri che costituiscono il programma, eventualmente preceduta da un'etichetta (label), che va messa tra parentesi graffe, e da commenti.

Le etichette si usano in due modi:

{etichetta} **<istruzione>** oppure
{etichetta} CONTINUE
<istruzione>

LE SUBROUTINE

All'interno di un programma «F-Basic» possono esserci due tipi di subroutine: il primo tipo è quello situato all'interno del programma principale, ovvero deve trovarsi prima dell'istruzione **END** finale. Queste subroutine vanno richiamate con il comando

GOSUB <nome>

dove *nome* è appunto l'etichetta con la quale viene contraddistinta la subroutine. Un esempio di questa struttura si trova in figura 1.

Questo tipo di subroutine deve sempre terminare con l'istruzione **RETURN** o **LRETURN**, e le variabili che si possono usare sono le stesse specificate nella parte dichiarativa.

Il secondo tipo di routine viene usato dopo l'istruzione **END**; in questo caso la routine viene richiamata con il suo nome o (se preceduta dall'istruzione **SUBROUTINE**), con il comando

CALL <nome>

Queste routine possono essere sottoprogrammi (le SUBROUTINE sopra accennate) o funzioni, strutturate entrambe allo stesso modo:

SUBROUTINE <nome>
<parte dichiarativa>
<parte eseguibile>
END

e

Fig. 4: Per visualizzare immagini grafiche in formato IFF non sono necessari sforzi particolari. Basta ricorrere al comando ReadIff.

```
PROGRAM MostraIff
INTEGER I
TEXT TT*1
OPEN (1,"Immagine.IFF")
I=SCREEN #2 (0,200,5,1,0)
I=WINDOW #1 (0,0,320,200,0,0,320,200,-1,-1,0,0,2)
I=READIFF(1,FILELENGTH("Immagine.IFF"),1)
TT=INCHAR()
CLOSE (1)
WINDOW_CLOSE #1
SCREEN_CLOSE #2
END
```

FUNCTION <nome
<parte dichiarativa>
<parte eseguibile>
END

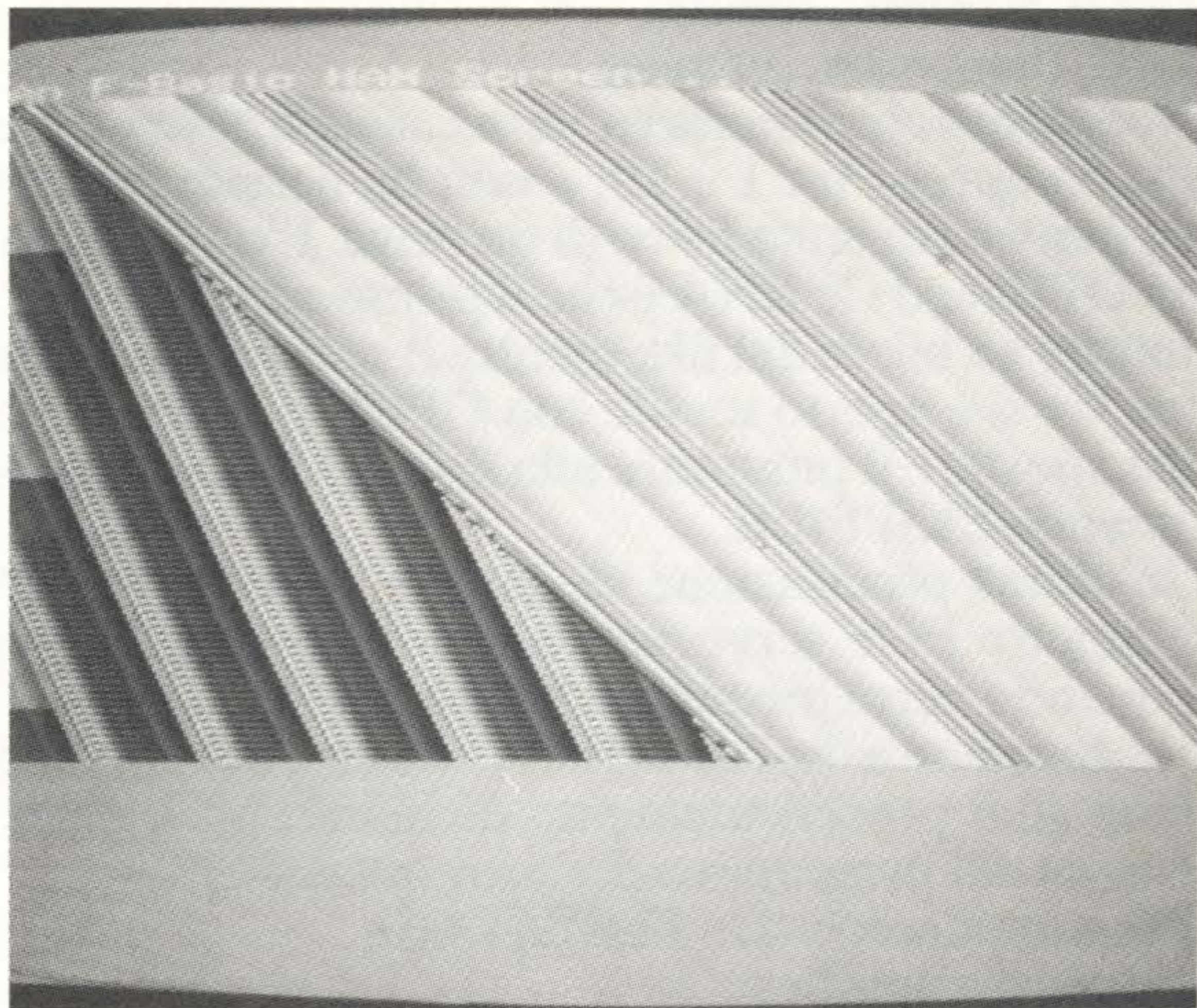
Le variabili che possono essere usate sono quelle dichiarate nella parte dichiarativa della subroutine, quelle passate come parametri, e le variabili precedentemente definite come globali. La parte dichiarative di una SUBROUTINE o di una FUNCTION ha questo formato:
PARAMETER
<dichiarazione parametri>
e
LOCAL
<dichiarazione parametri locali>
In figura 2 potete vedere un esempio: il risultato del listato sarà la stampa su video dei caratteri «CIAO 5» (stampati dalla subroutine CIAO) e dei caratteri «12» (PRINT Funzione (*X*)). Noterete che quando si passano dei parametri, il loro valore viene «scaricato» nella corrispondente variabile dichiarata con PARAMETER (quello che c'è in *X* con Funzione (*X*) viene messo in *Z*).

INPUT E OUTPUT

In «F-Basic» i comandi **INPUT** e **PRINT** si comportano in maniera simile agli altri Basic. L'unica differenza consiste nell'assenza del punto e virgola per non far andare a capo il cursore. Le variabili da stampare infatti devono essere separate solo con la virgola, che lascerà solo uno spazio tra i valori.

Non esiste invece l'istruzione **PRINT USING**; per formattare l'output delle variabili la sintassi è un'altra:
PRINT <variabile>[<formato>]
dove *formato* è, nel caso di numeri interi o di stringhe, il numero di caratteri da stampare, mentre nel caso di numeri reali è un'espressione così strutturata:
[E o F<#num_sin>,<#num_des>]
dove *E* o *F* impostano rispettivamente la notazione esponenziale o decimale, *num_sin* è il numero di cifre da visualizzare a sinistra della virgola, e **num_des** quello di cifre da visualizzare a destra. Per stampare spazi è disponibile la funzione **SPACE(n)**, che produce un numero **n** di spazi.

Il controllo del cursore e dell'output su schermo è svolto da numerosi comandi, il cui elenco è riportato in figura 3; spendiamo due parole solo sulla funzione **CURS_POS()**: per usarla, occorre eguagliarla ad una variabile, ma in un modo un po' diverso dal solito:
X = HIWORD(CURS_POS())
Y = LOWORD(CURS_POS())
Al posto di *X* ed *Y* andranno specificate la riga e la colonna del cursore. Questo metodo si usa in quanto il valore ritornato da CURS_POS() è una longword la cui parte alta contiene il valore della riga del cursore, mentre quella bassa contiene quello della colonna. **HIWORD()** e **LOWORD()** sono appunto funzioni che restituiscono rispettivamente la word alta e bassa di una longword.

Le operazioni di input ed output relative a file in «F-Basic» sono controllate dai seguenti comandi:
OPEN (num_canale, nome)
OPENA (num_canale, nome)
INPUT #<num_canale> EOF=<etichetta>
<espressioni>

I COMANDI GRAFICI

Questo elenco racchiude alcuni tra i più semplici comandi grafici disponibili in «F-Basic»:
COLOR_DEFINE #<colore> (rosso, verde, blu): *Colore* dipende dal numero di bitplane usati: se lo schermo ha 4 bitplane, il valore di *colore* può essere compreso tra 0 e 15.
COLOR_PENS (<colore testo>, <colore sfondo>): Definisce i colori usati per l'output di testi.
COLOR_POINT #<num_colore> (x,y): Colora un pixel; con COLOR_RPOINT le coordinate sono relative.
<variabile> = COLOR_READ (x,y) o <variabile> = COLOR_RREAD (x,y): Legge il colore di un pixel.
COLOR_LINE #<num. del colore> (<modo>, x1, y1, x2, y2): Disegna una linea. Il parametro *modo* è un numero da 0 a 3 con i seguenti significati:
0 = prima coordinata assoluta, seconda assoluta
1 = prima coordinata relativa, seconda assoluta
2 = prima coordinata assoluta, seconda relativa
3 = prima coordinata relativa, seconda relativa
COLOR_BOX #<num. del colore> (<modo>, x1, y1, x2, y2): Disegna un rettangolo. I parametri sono gli stessi di COLOR_LINE.
COLOR_BOXFILL #<num. del colore> (<modo>, x1, y1, x2, y2): Come COLOR_BOX, ma riempie il rettangolo.
COLOR_ELLIPSE #<num. del colore> (xcen, ycen, xrad, yrad) o COLOR_RELLIPSE #<num. del colore> (xcen, ycen, xrad, yrad): Disegna un'ellisse. Le coordinate *xcen, ycen* si riferiscono al centro dell'ellisse, mentre *xrad, yrad* sono rispettivamente il raggio orizzontale e quello verticale
COLOR_POLYDRAW #<colore> (<array>, <num. coppie>): Disegna un poligono. Il parametro *array* è un puntatore ad un array contenente le coordinate degli spigoli del poligono e *coppie* indica il numero di coppie di coordinate, cioè il numero di spigoli.
COLOR_FLOOD #<num. del colore> (x, y, <colorebordo>) o COLOR_RFLOOD #<num. del colore> (x, y, <linea>): Colora l'area intorno a *x, y* in tutte le direzioni, fermandosi in una direzione quando trova una linea del colore *linea.*
COLOR_AREA (x, y) o COLOR_RAREA (x, y): Setta le coordinate *x, y* come uno spigolo di un poligono. Una sequenza di questi comandi ha lo stesso effetto di COLOR_POLYDRAW.
COLOR_AREAFILL: Riempie il poligono segnato con una sequenza di COLOR_AREA o COLOR_RAREA.

PRINT # <num_canale> <espressioni>
SKIPFILE (num_canale)
CLOSE (lista num_canale)

Precisiamo che il numero *num_canale* deve essere un intero compreso tra 1 e 10, e non ha nessuna attinenza con il numero di unità fisico di un drive. L'unica differenza tra OPEN ed OPENA è che il primo comando lascia il puntatore all'inizio del file, mentre il secondo lo porta alla fine. Se si è aperto un file con OPEN e si vuole ugualmente spostare il puntatore alla fine del file, si può usare il comando SKIPFILE.

Un'ultima precisazione va data sul comando INPUT: il parametro EOF serve per indicare l'etichetta dove saltare quando verrà incontrata la fine del file.

Per accedere a dei record in «F-Basic», occorre che essi siano stati precedentemente dichiarati nella parte dichiarativa del programma. La loro sintassi è:

TYPE <nomerecord> IS RECORD
<dichiarazioni>
ENDTYPE

dove *dichiarazioni* è una lista di questo genere:

<tipo var.> <var1>, ..., <varN>

Per accedere ad un elemento del record bisogna dichiarare delle variabili di tipo *nomerecord* e poi riferirsi ad esse in questo modo:

<nome_var>,<nome_campo>

dove *nome_campo* è il nome di una variabile della parte *dichiarazioni* di una struttura TYPE...IS RECORD/ENDTYPE. Il tipo di un campo di un record è quello dichiarato in *dichiarazioni*.

Infine, «F-Basic» permette l'uso di puntatori: per usarli occorre dichiarare delle variabili di tipo **PTR_TO.** Il modo più semplice per assegnare dei valori ad un puntatore è uguagliarlo con la funzione @**<variabile>** che ritornerà l'indirizzo della variabile specificata. La funzione @ comunque, in altre istruzioni, accetta anche valori diversi: ad esempio @"CIAO" restituirà l'indirizzo dei caratteri «ciao».

LA GRAFICA

Vediamo ora come aprire uno schermo. La sintassi è la seguente:

variabile = SCREEN # <num> (attributi)

dove *variabile* è un numero intero che identifica il puntatore alla struttura dello schermo appena aperto, e *num* è il numero dello schermo (può essere compreso tra 1 e 4). Gli *attributi* dello schermo devono essere, nell'ordine: inizio, altezza, profondità, modo e titolo (separati da virgole).

Inizio è la coordinata verticale espressa in pixel da cui partirà lo schermo; *altezza* rappresenta l'altezza, sempre in pixel, dello schermo; *profondità* rappresenta il numero di bitplane da attivare; *modo* è un numero che identifica la risoluzione e *titolo* è il puntatore ad una stringa contenente il titolo dello schermo (usare 0 per omettere il titolo). Il parametro *modo* può assumere i seguenti valori:

1 = bassa risoluzione, non interlacciata;
2 = alta risoluzione, non interlacciata;
3 = bassa risoluzione, interlacciata;
4 = alta risoluzione, interlacciata.

Per aprire una finestra si usa la funzione **WINDOW:**

variabile=WINDOW # <num> (<attributi>)

In *variabile* verrà posto il puntatore alla struttura della finestra. Il parametro *num* deve essere il numero della finestra (da 1 a 20), mentre i possibili *attributi* sono:

x1, y1, x, y, minx, miny, maxx, maxy, detailpen, blockpen, flags, titolo, num

dove *x1* e *y1* sono le coordinate dell'angolo in alto a sinistra; *x* e *y* sono rispettivamente l'ampiezza e l'altezza; *minx, miny, maxx* e *maxy* sono i valori minimi e massimi delle dimensioni; *detailpen* e *blockpen* sono i colori da usare per colorare la finestra (i colori di default sono −1 e −1); *flags* è la somma dei seguenti attributi:

0 = SimpleRefresh
1 = Sizing Gadget
2 = Drag Bar
4 = Depth Gadget
8 = Close Window Gadget
16 = SmartRefresh.

Per il loro significato potete consultare il manuale di AmigaBasic nella parte dedicata al comando WINDOW. Infine, *titolo* è il puntatore ad una stringa contenete il titolo della finestra, e *num* è il numero dello schermo nel quale deve essere aperta la finestra.

```
ByteCalc
PROGRAM BYTECalcBenchmark
REAL C,B,A
INTEGER I,N
DATA (N,5000),(A,2.781281828),(B,3.141592654),(C,1.0)
TYPE DateStamp IS RECORD
   INTEGER DAYS,MINUTES,TICKS
ENDTYPE
DateStamp AA,BB
SUBPROGRAM
   SUBROUTINE ElapsedSeconds
I=DOSTIME(@AA)
FOR I=1 TO N
   C=C*A ; C=C*B ; C=C/A ; C=C/B
NEXT I
I=DOSTIME(@BB)
PRINT "Tempo impiegato:",
ElapsedSeconds(@AA,@BB)
END
SUBROUTINE ElapsedSeconds
PARAMETER
   PTR_TO DateStamp TBEGIN,TEND
LOCAL
   INTEGER TotSeconds
&QUICK 0
TotSeconds=0
TotSeconds=(TEND.DAYS-TBEGIN.DAYS)*4320000+(TEND.MINUTES-TBEGIN.MINUTES)
          *3000+(TEND.TICKS-TBEGIN.TICKS)
PRINT TotSeconds/(50.0)
GRETURN
END
```

Il classico benchmark di Byte per calcolare la velocità di calcolo del compilatore. «F-Basic» lo esegue 20 volte più velocemente di AmigaBasic!

Per la gestione delle finestre esistono i seguenti comandi:

WINDOW_ACTIVE #n: attiva la finestra *n*;
WINDOW_BACK #n: porta la finestra *n* in fondo;
WINDOW_CLOSE #n: chiude la finestra *n*;
WINDOW_FRONT #n: porta avanti la finestra *n*;
WINDOW_OUTPUT #n: abilita la finestra *n* come output;
WINDOW_SIZE #n (x,y): cambia dimensioni alla finestra *n*;
WINDOW_MOVE #n (x,y): muove la finestra *n*;

Negli ultimi due comandi le coordinate *x* ed *y* sono relative, ovvero si riferiscono alla posizione attuale della finestra. Ad esempio:

WINDOW_SIZE #1 (1,-10)

cambierà le dimensioni della finestra numero 1, aumentando la larghezza di 1 e diminuendo l'altezza di 10.

Per avere informazioni sullo stato delle finestre esiste la funzione **WINDOW_INFO** la cui sintassi è:

<variabile>=WINDOW_INFO(num)

dove **num** può essere un numero da 1 a 11. I valori restituiti dalla funzione sono numeri interi, mentre il valore di *num* determina il tipo di informazione richiesto alla fun-

zione, secondo la seguente tabella:
1 = numero della finestra corrente (ovvero quella attualmente abilitata all'output).
2 = indirizzo della struttura dati dalla finestra corrente.
3 = indirizzo della struttura Rastport della finestra corrente.
4 e **5** = larghezza ed altezza della finestra attuale.
6 e **7** = coordinate x ed y della drawing pen.
8 e **9** = coordinate x ed y dell'angolo in alto a sinistra.
10 e **11** = coordinate x ed y del mouse.

LA GESTIONE DEI MENU

I tradizionali menu a discesa di Amiga sono gestiti in «F-Basic» tramite questa funzione:
<variabile>=MENU #<num> (<text>, <elem>)
Essa viene usata per settare le opzioni nei menu. Il parametro *num* è il numero del menu su cui operare (da 1 a 10), *text* è il puntatore ad un array contenente gli elementi del menu ed *elem* è il numero di elementi.
Le altre principali istruzioni relative ai menu sono:
MENU_DISABLE #<num> o **MENU_DISABLE #<num> (elem)**: disabilita la voce *num* o l'elemento *elem*;
MENU_ENABLE #<num> ≈ **MENU_ENABLE #<num> (elem)**: abilita una voce od un elemento;
MENU_CHECK #<num> (elem): Mette il cosidetto «checkmark» all'elemento *elem*.
Per controllare i menu o gli altri input dell'utente, «F-Basic» mette a disposizione un potente comando per la gestione degli eventi:
ON <evento> EVENT
<istruzioni>
END EVENT
dove *evento* può essere:
WINDOW_CLOSE: la chiusura di una finestra;
SINGLE_CLICK: un singolo click del pulsante sinistro del mouse;
DOUBLE_CLICK: due click consecutivi del pulsante del mouse;
MENU_SELECT: la selezione di una voce di un menu;
INKEY: la pressione di un tasto.
Una volta incontrato il comando ON...EVENT, ogniqualvolta avverrà l'evento specificato verranno eseguite le istruzioni indicate nel ciclo.

Una nota finale su come utilizzare le librerie di sistema in «F-Basic»: come già precedentemente accennato, bisogna innanzitutto dire al compilatore che si intende accedere alle librerie di sistema con la direttiva &SYSLIB ed aprire la libreria desiderata nel programma con la funzione
<variabile>=OPENLIB("<nome>",<min_ver>)
che ritornerà il puntatore alla libreria in *variabile* (il valore sarà 0 se si riscontra un errore). Il parametro *min_ver* indica il numero minimo di versione richiesto.

Per richiamare i comandi o le funzioni delle librerie la sintassi è la seguente:
<comando>(<punt>,<par1>,...,<parN> oppure **<var>=<funzione>(<punt>,(par1>,...,<parN>)**
dove *punt* è il puntatore alla libreria e *par1 ... parN* sono i parametri necessari.

□

AUDIOMASTER III

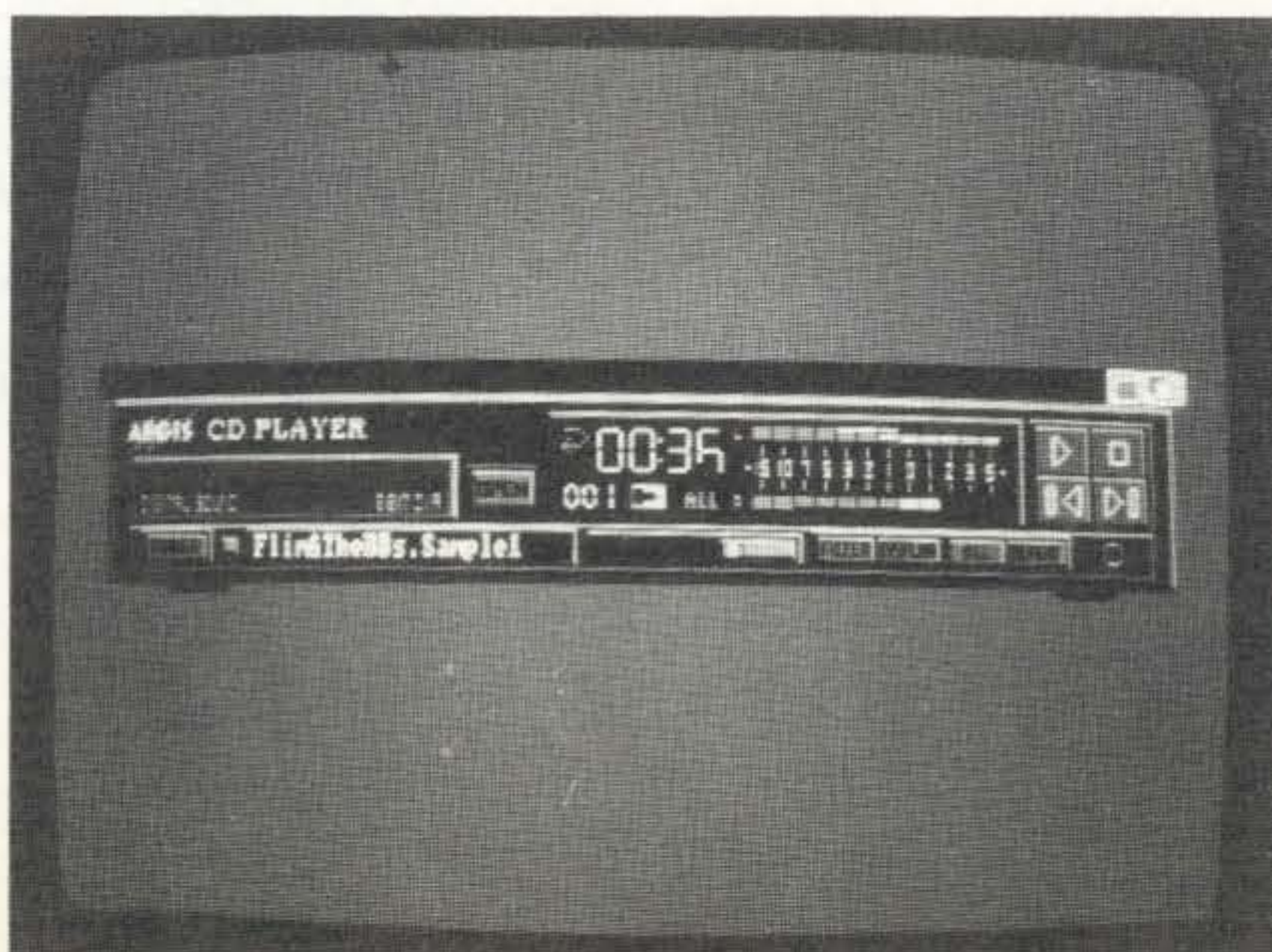
di Graziano Pavone

Acquistando un' Amiga si rimane subito stupiti per le sue incredibili capacità sonore, sicuramete le migliori tra quelle dei computer della sua stessa fascia di prezzo.

Per coloro che usano il proprio Amiga come sequencer, collegandolo via MIDI con altre apparecchiature musicali, per coloro che sono appassionati di musica o che semplicemente vogliono sfruttare a fondo le capacità sonore del proprio computer, è molto interessante, quindi, la possibilità di campionare suoni provenienti da una qualsiasi sorgente, per mezzo di un apposito digitalizzatore audio.

In commercio vi è più di un apparecchio di questo tipo, tutti molto diversi tra loro per prezzo e per fedeltà di campionamento, apparecchi che possono essere utilizzati con diversi programmi che consentono di digitalizzare suoni e di eseguire su di essi funzioni di editing più o meno potenti.

«**AudioMaster III**» della Oxxi/Aegis, è uno di essi. Grazie alle modifiche apportate in questa recente versione, esso può essere utilizzato in unione a qualsiasi digitalizzatore audio. Tramite la voce «**Sampler Config** ...» del **menu** «**Options**», infatti, è possibile scegliere tra diverse configurazioni, così che alcuni apparecchi precedentemente non supportati (come il campionatore del pacchetto «**Pro Sound Designer**») funzionano ora perfettamente. È possibile inoltre inserire manualmente i valori della frequenza massima di campionamento.

Altre caratteristiche molto importanti del programma sono la sua intuitività e semplicità d'utilizzo, dovuta al fatto che tutte le operazioni possono essere eseguite con il solo ausilio del mouse.

La forma d'onda appare infatti sullo schermo in un apposito riquadro (o in due se si tratta di un suono stereo) e può essere manipolata semplicemente selezionando una parte di essa con il mouse e scegliendo la funzione desiderata: con la **funzione «Show Range»**, ad esempio, si può ingrandire la parte selezionata fino a visualizzare sullo schermo soli 599 byte di suono campionato, e ci si può spostare ovunque fino a trovare la porzione di suono che ci interessa agendo semplicemente, sempre con il puntatore del mouse, sulla comoda barra orizzontale posizionata sotto il riquadro riservato alla visualizzazione della forma d'onda.

Passiamo ora ad esaminare in dettaglio le funzioni del programma: il primo menu, chiamato come al solito «**Project**», consente di caricare un suono da disco o addirittura di cercarne uno nella Ram del computer, e di salvare quello in memoria su supporto magnetico. Il salvataggio può avvenire in formato standard IFF o, cosa utilissima, in uno speciale formato compresso, introdotto con questa nuova versione del programma, che consente di dimezzare lo spazio necessario.

La voce «**SnapShot**», invece, consente di salvare la forma d'onda nella directory corrente in un file temporaneo che potrà poi essere richiamato tramite la **funzione «Recall»**, per evitare di perdere il lavoro fatto.

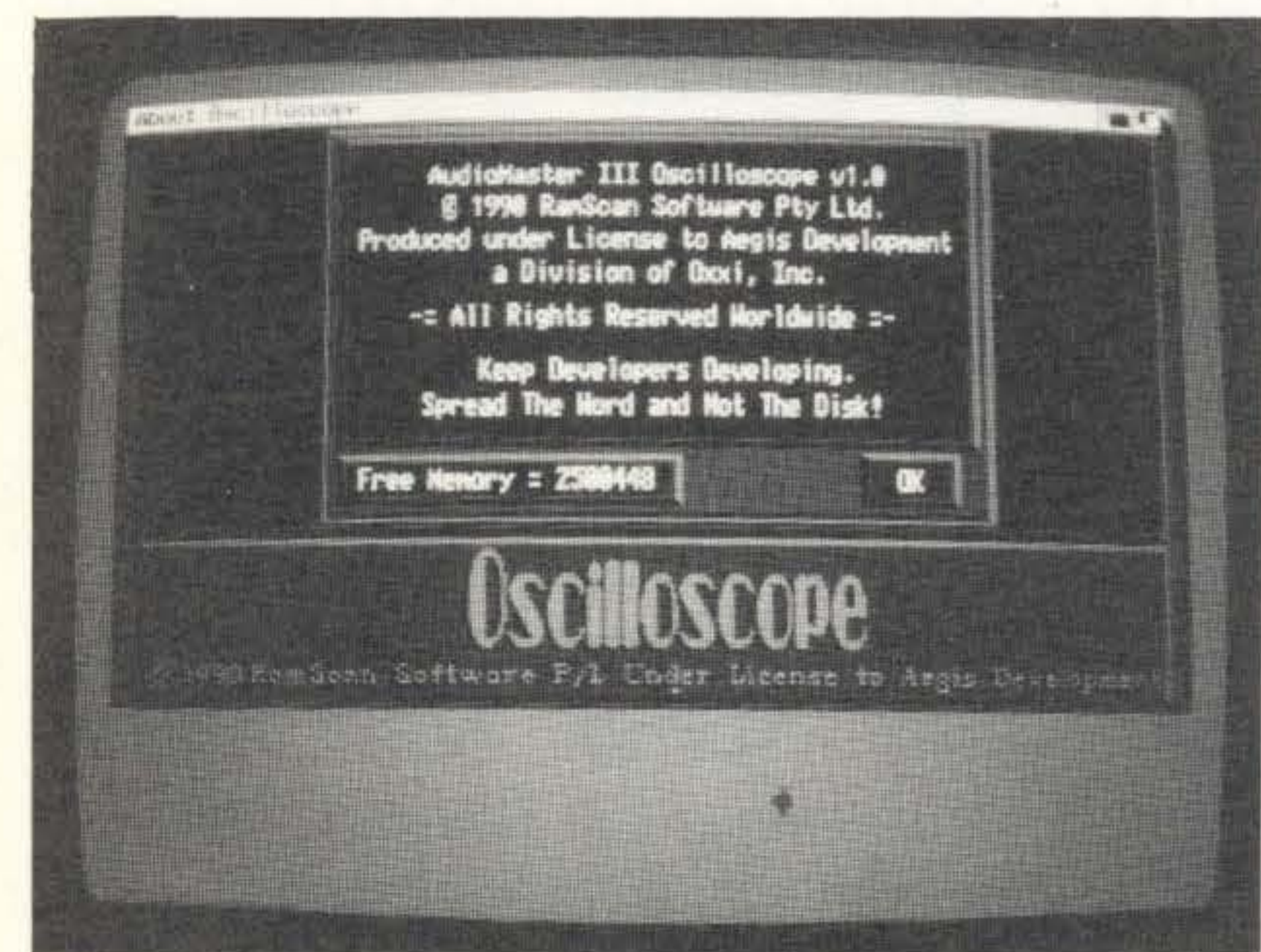
Per campionare un suono bisogna per prima cosa decidere, tramite la voce «**Mode ...**» del menu «Project», se il suono dovrà essere stereofonico o meno, tenendo conto delle proprie esigenze e delle possibilità del proprio hardware, poiché la fedeltà con la quale il suono viene campionato è direttamente proporzionale alla memoria necessaria.

Dopo questa operazione occorre selezionare l'**opzione «Sampler»** dallo stesso menu. Apparirà una finestra chiamata «**Digital Sampler**» contenente due barre: una per informare il programma della memoria da utilizzare, l'altra per decidere a quale frequenza, nei limiti di quella massima consentita dal digitalizzatore, i suoni dovranno essere campionati.

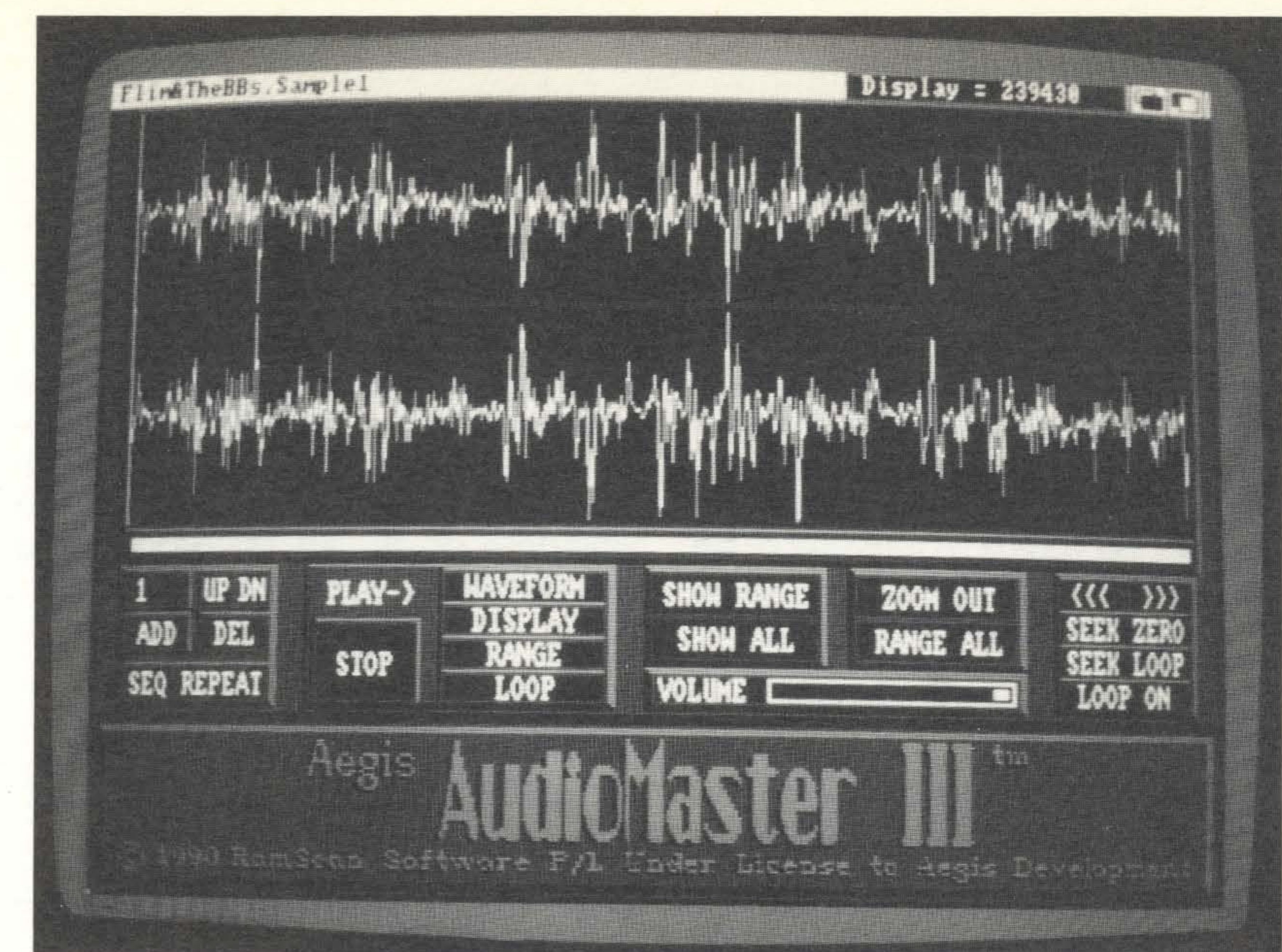
Vi sono poi tre gadget in basso che consentono di controllare il segnale in ingresso per regolare di conseguenza il volume («**Monitor**»),

di scegliere se il campionamento dovrà iniziare quando il segnale raggiungerà il volume indicato nella finestra di configurazione («**Vox On**») o manualmente alla pressione del tasto del mouse («**Vox Off**»), e di iniziare il campionamento vero e propio («**Sample**»).

Dopo aver fatto ciò, «AudioMaster» ci mette a disposizione, nel **menu «Edit»**, oltre che le funzioni di editing standard di taglia/incolla (cioè le solite **cut**, **copy** e **paste** utilizzate anche nei word processor e quindi sicuramente conosciute da tutti), anche funzioni più complesse quali «**Invert**», che inverte la fase dei dati, «**Swap Chan**», che scambia il canale destro e quello sinistro in un suono stereo; «**Zero**» e «**Edit Freehand**», invece, consentono di azzerare una parte della forma d'onda o di disegnarla direttamente sullo schermo con il mouse.

Ci sono poi funzioni ancora più potenti nel **menu «S/FX»**, che agiscono sul valore dei singoli byte e che consentono di aggiungere un effetto di eco («**Echo**»), di invertire l'ordine dei dati facendo riprodurre

il suono dalla fine all'inizio («**Backwards**»), di miscelare due diversi suoni («**Mix Waveforms**»), di cambiare il volume o aggiungere effetti di dissolvenza («**Change Volume**»), o di alterare la velocità di riproduzione, ed eventualmente simulare un campionamento ad una frequenza diversa da quella reale («**Tune Waveform**»).

Tramite «**Low Pass Filter**», infine, è possibile inserire un filtro passa basso software per eliminare, o almeno attutire, la rumorosità delle forme d'onda.

In aggiunta a questi effetti applicabili a ciò che si è campionato, vi sono due effetti che agiscono direttamente sul segnale in ingresso: «**Real Time Echo**» e «**Real Time Flange**», che producono un'echo o una distorsione del suono, secondo parametri da impostare comodamente con il mouse.

Il punto di forza di «AudioMaster III», la cosa che lo distingue dalla versione precedente e da tutti gli altri programmi per il campionamento di suoni per Amiga, è la **funzione multi-loop**, che rappresenta un grosso passo in avanti e che avvicina le possibilità offerte dal programma a quelle dei campionatori dedicati.

Il programma consente cioè di settare più di una coppia di puntatori per i loop e quindi di gestire, in fase di esecuzione, ripetizioni di segmenti diversi della forma d'onda, permettendo di creare suoni molto complessi e di arrivare ad un livello veramente professionale.

Tutto ciò è possibile grazie ad un menu dal quale non abbiamo ancora parlato, «**Sequence**», ed all'insieme di gadget collocati nell'angolo in basso a sinistra dello schermo. Selezionando il **gadget «Seq»**, il menu verrà attivato e sarà quindi possibile cominciare ad editare la **sequence**, cioè la lista di loop che dovranno essere eseguiti.

Dopo aver impostato il punto di partenza e quello d'arrivo del primo loop, bisogna selezionare il **gadget «Add»**, che aggiunge un loop alla sequenza, e fare lo stesso per il secondo e per tutti i seguenti.

Quando si sarà terminato il proprio lavoro, ci si potrà muovere all'interno della sequenza usando i **gadget «Up»** e «**Dn**», oppure si potrà cancellare un loop con il **gadget «Del»**.

Il selezionamento di «**Repeat**», invece, consente che la sequenza sia ripetuta in continuazione fino alla pressione di «**Stop**» o della barra spaziatrice.

Per l'editing della **sequence** è importante inoltre l'apposito menu che consente di eseguire operazioni di taglia-incolla. Molto utili ed interessanti sono soprattutto le **funzioni** «**Set Fade Start**» e «**Strip Dead Space**». La prima fa sì che il volume si abbassi, a partire dal loop selezionato fino ad annullarsi completamente, creando ottimi effetti di dissolvenza; la seconda elimina le parti della forma d'onda che non vengono utilizzate dalla **sequence**, con lo scopo di risparmiare preziosa memoria.

L'unico svantaggio derivante dall'utilizzare la tenica del **multi-loop** è che essa non può essere adoperata con altri programmi di riproduzione di suoni campionati, ma solo con quelli forniti insieme ad «AudioMaster»; è probabile però che in futuro anche altre software house aggiornino i propri programmi ed implementino in essi il supporto del multi-loop.

Nel pacchetto originale sono presenti infatti, oltre che «AudioMaster III», altri programmi: «CD Player Simulator», «Oscilloscope», «Multimaker», e «MergeIFF».

«**CD Player Simulator**» è, come dice il nome stesso, un simulatore di lettore di compact disc: specificando al programma una serie di nomi di file di suoni campionati IFF (al massimo 20), potremo gestirli come fossero brani di un CD, agendo con il mouse su di un'immagine molto ben curata rappresentante il frontale di un lettore di CD.

Il secondo programma, «**Oscilloscope**», ha invece l'unico scopo di visualizzare il segnale in ingresso, per regolare il volume con maggior precisione, mentre gli altri due ser-

vono rispettivamente a miscelare alcuni suoni per generare un unico file e ad unire, una di seguito all'altra, due forme d'onda.

Molto più utili risultano «**PlaySound**», che permette di suonare un file generato da «Audio Master» eseguendo anche eventuali loop multipli, e «**MakeStereo**», che unisce due file mono in un unico file stereofonico. Non manca infine un comando che converte file IFF in files RAW, cioè in semplici sequenze di byte rappresentanti il suono digitalizzato.

«AudioMaster III» è senza dubbio il miglior programma per il campionamento di suoni per Amiga: l'ottima fedeltà di riproduzione e le potenti capacità di editing della versione precedente sono state ulteriormente migliorate, la incompatibilità con alcuni digitalizzatori è stata eliminata, ma soprattutto a tutto ciò si è aggiunta la capacità di gestire più di un loop. Cosa si può chiedere di più ad un programma del genere?

OXXI/AEGIS
P.o. Box 90309
Long Beach, CA
90809-0309 USA

ZX SPECTRUM EMULATOR

di Luca Brigatti

Fra i computer che hanno fatto la storia della microinformatica un posto di rilievo spetta indubbiamente allo **ZX Spectrum**. Immesso sul mercato nel 1982 (solo nove anni fa, anche se sembra un'altra era) da parte della Sinclair Ltd., di proprietà del geniale inventore Sir Clive Sinclair, rappresentò una rivoluzione nel suo genere.

Computer dal costo molto contenuto e dall'aspetto insolito, offriva prestazioni notevoli per l'epoca (ad esempio una memoria RAM di ben 48 Kbyte!) eguagliate qualche tempo dopo dal Commodore 64.

Molti degli attuali utenti Amiga, soprattutto i più «hacker» e smanettoni, hanno imparato a programmare proprio sullo Spectrum. Forse è

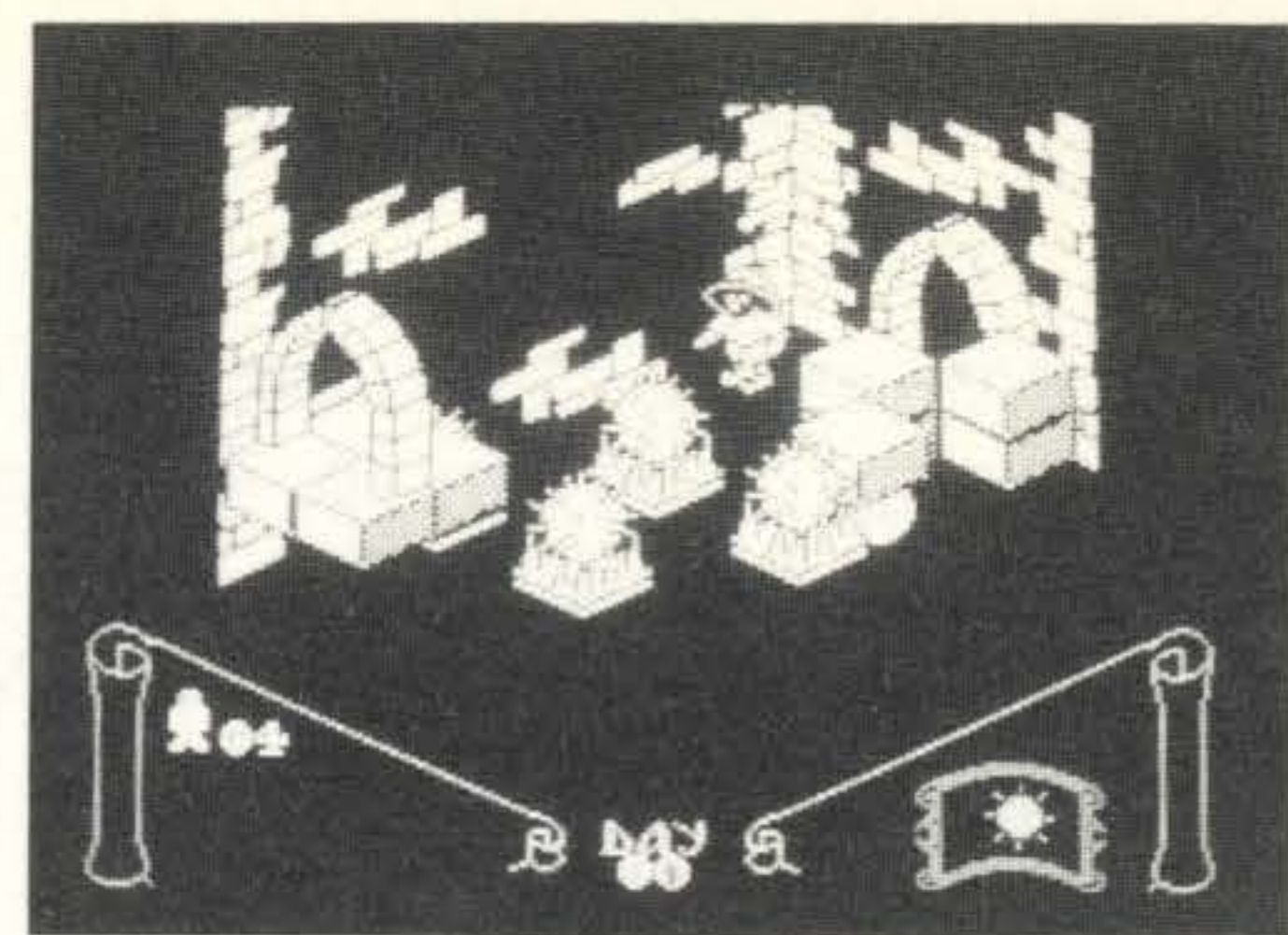
per questo che, nonostante la presenza sul mercato di computer dalle prestazioni incommensurabilmente migliori, primo fra tutti l'Amiga, un giovane programmatore toscano, Vincenzo Gervasi, ha deciso di realizzare un programma in grado di emulare lo Spectrum.

Per coloro che non fossero già «passati» attraverso lo Spectrum ma ne avessero sentito narrare le gesta dagli «anziani» (ricordiamo comunque che lo Spectrum è tutt'ora in commercio sotto marchio Amstrad), riassumiamo le caratteristiche salienti della macchina, presenti anche nel programma di emulazione.

Lo ZX Spectrum utilizza il microprocessore **Z80** della Zilog a 8 bit; possiede una memoria **RAM** di **48 Kbyte** (o in alternativa di **16 Kbyte**, nelle prime macchine) ed una **ROM** con il Basic incorporato, che contiene anche l'ambiente di lavoro standard dello Spectrum.

La capacità grafica vanta **8 colori**: bianco, nero, rosso, verde, blu, giallo, magenta ed azzurro, con due varianti di luminosità. I colori dello sfondo (**Paper**), del bordo (**Border**) e delle scritte (**Ink**) sono indipendenti

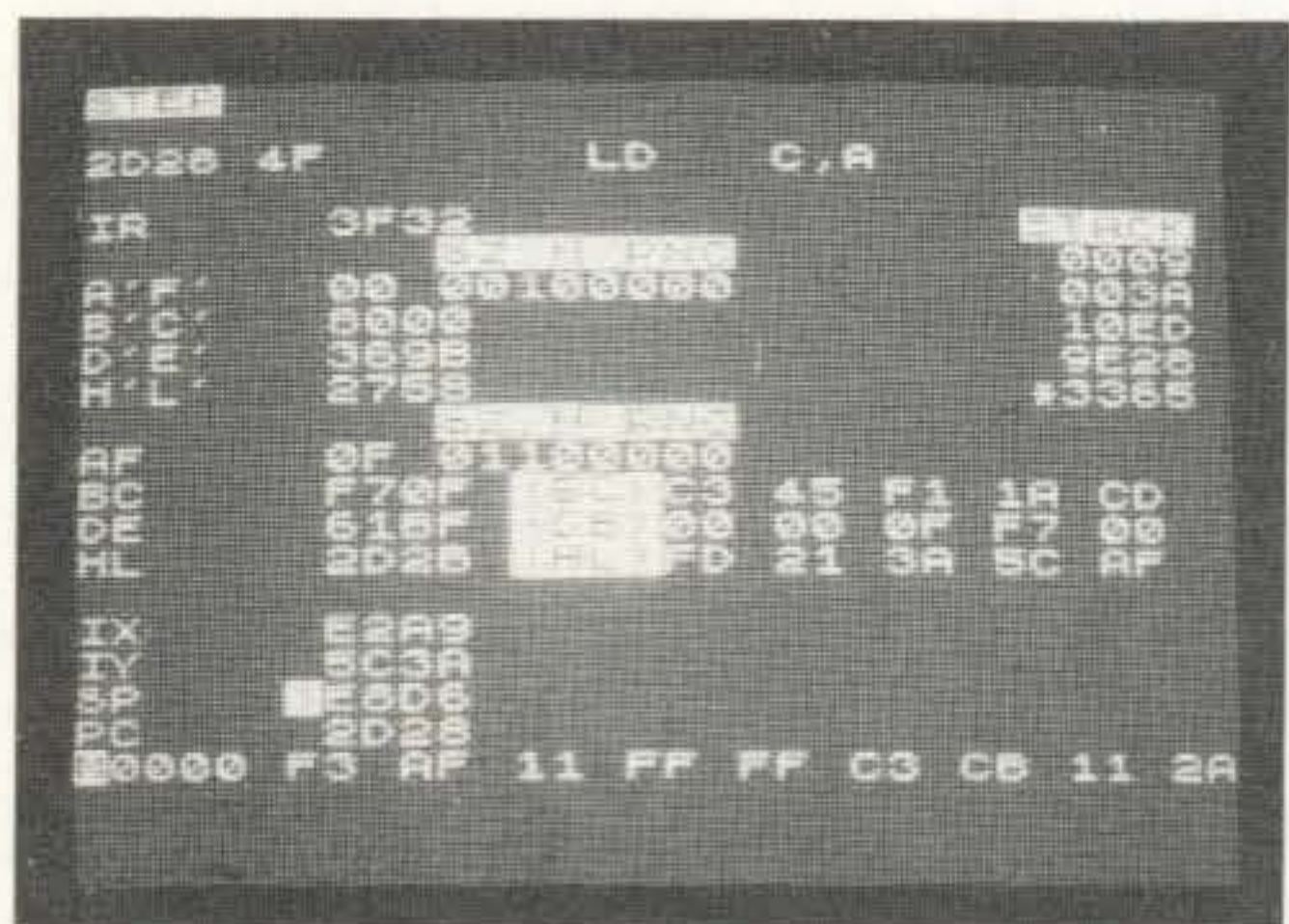
Inoltre ogni singolo carattere può avere un colore del tratto e dello sfondo diverso da quelli adiacenti.

La risoluzione grafica è di

256×176 pixel indirizzabili singolarmente. I pixel possono avere colore diverso ma devono avere lo stesso colore nell'ambito di una matrice di 8×8 (lo spazio di un carattere). Lo Spectrum possiede un altoparlante piezoelettrico incorporato in grado di emettere un'ampia gamma di note su di un solo canale. Tutte queste caratteristiche grafiche e sonore vengono fedelmente riprodotte durante l'emulazione con Amiga. In particolare il suono è supportato dall'emulatore con una resa decisamente superiore a quella dell'altoparlante piezoelettrico dello Spectrum.

Lo «**ZX Spectrum Emulator**» viene fornito con un manuale per l'utente ed un manuale per il programmatore. È acclusa al software, inoltre, un'interfaccia hardware in grado di collegare all'Amiga un registratore a cassette per il Load e per il Save dei programmi originali dello Spectrum; è inoltre prevista anche la possibilità di accedere alle funzioni di AmigaDos durante l'emulazione, e quindi di salvare o caricare file Spectrum da dischetto. Il formato di memorizzazione su nastro è però quello Spectrum standard: i vari formati «**Turbo**» per il caricamento accelerato da nastro non sono supportati dall'emulatore.

Dopo l'accensione di Amiga si inserisce il dischetto sul quale si trova il programma di emulazione, che carica il suo WorkBench. A questo punto basta aprire la finestra del dischetto e clickare sull'icona che rappresenta la sagoma familiare dello Spectrum.

A caricamento avvenuto si entra nel tradizionale ambiente di lavoro Spectrum con il suo schermo nero con bordo bianco che appare durante un Reset. Dopo qualche secondo (a reset avvenuto) appare lo schermo bianco con la tradizionale scritta «**(c) 1982 Sinclair Research Ltd.**», e, quasi per magia, ci si ritrova a lavorare non più con un Amiga bensì con uno Spectrum.

La caratteristica più sconcertante, e nello stesso tempo più comoda per i programmatori dello Spectrum, consiste nel fatto che tutti i comandi e le funzioni sono richiamabili, già

scritte per esteso, premendo un tasto o una combinazione di due o più tasti. Non si tratta di una caratteristica facoltativa: i comandi e le funzioni non si possono in alcun modo scrivere battendo i singoli tasti del loro nome.

Questa peculiarità è rimasta sull'emulatore; così, per esempio, per il **comando Print** basta premere il **tasto P**, per **For** il **tasto F**, per **Next** il **tasto N**, e così di seguito.

Per richiamare alcuni altri comandi o funzioni occorre premere uno dei due **tasti Shift**, denominato **Symbol Shift** e rappresentato sulla tastiera dell'Amiga dal **tasto ALT**.

Il tasto Shift che rende maiuscoli i caratteri viene definito **Caps Shift** e corrisponde al normale **Shift**.

Per richiamare altri comandi è necessario entrare in modo esteso (**Extended Mode**) premendo contemporaneamente i due tasti Shift oppure, come avviene per lo **Spectrum Plus**, premendo il **tasto Extended Mode** che sulla tastiera dell'Amiga corrisponde al **tasto Tab**. Per entrare nel modo Grafico, che consente di stampare gli otto caratteri grafici presenti sui tasti numerici da 1 a 8, basta premere **Caps Shift-9.**

Altre aggiunte dell'emulatore sono la presenza di un **tasto Break**, alternativo alla tradizionale combinazione **Caps Shift-Spazio** che corrisponde al tasto «~» (in alto a destra, sotto Esc); e la presenza di un **tasto Edit** (per richiamare nella parte bassa dello schermo una riga di programma e modificarla) alternativo alla combinazione **Caps Shift-1** e corrispondente al **tasto Esc** di Amiga.

Un'altra importante variante introdotta con l'emulatore è la possibilità di utilizzare i tasti freccia di Amiga per muovere il cursore verticalmente lungo il listato, oppure orizzontalmente lungo una linea in fase di Editing, senza la necessità di utilizzare la combinazione di Caps Sfift con i tasti numerici da 5 ad 8.

Sempre per quanto riguarda la tastiera, il funzionamento del **tasto Del** corrisponde a quello del **tasto Backspace**. Tutti i principali comandi da tastiera sono riassunti in un **help**, richiamabile sullo schermo in qualsiasi momento durante l'emulazione.

Dell'emulatore è possibile pilotare la stampante utilizzando normalmente i comandi **Lprint**, **Llist** e **Copy**; il supporto della stampa è totale, sia tramite il Basic che direttamente da linguaggio macchina.

Caratteristiche proprie dell'emulatore, non presenti originariamente sullo Spectrum, sono la possibilità di utilizzare direttamente un normale joystick di Amiga in porta 2 (mentre non è naturalmente possibile utilizzare il mouse, che non esisteva ai tempi di Sir Clive) e la capacità di riconoscere il **joystick Kempston**, quello che va per la maggiore con lo Spectrum.

Altra caratteristica interessante è la capacità di leggere e di scrivere file utilizzando i drive di Amiga in modo AmigaDos (senza ulteriori complicazioni) nonché di accedere direttamente all'AmigaDos dall'ambiente Spectrum.

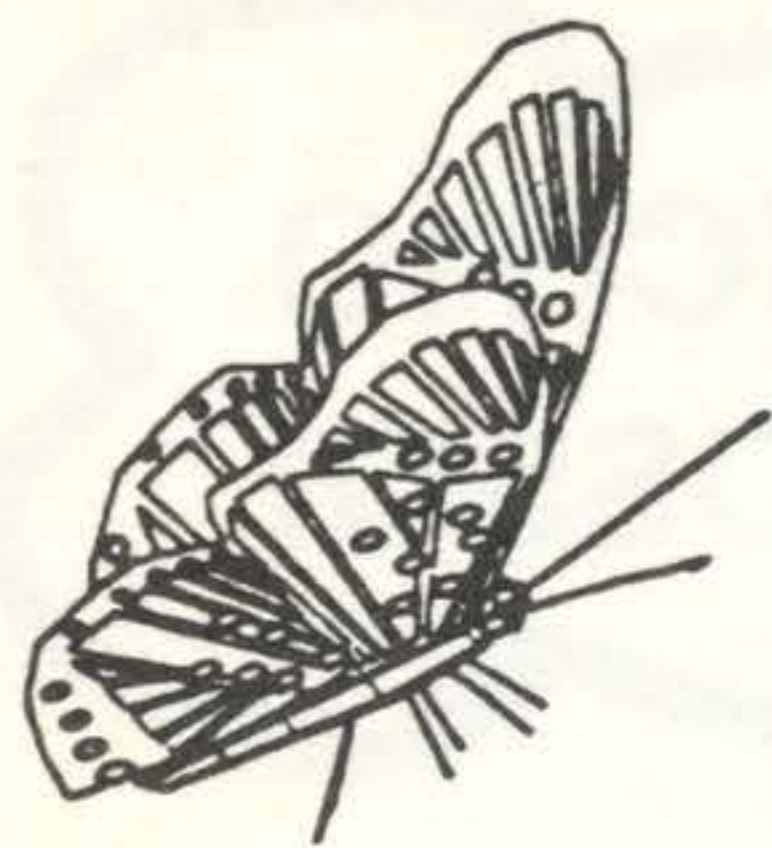
Per ragioni di velocità, è stata aggiunta la possibilità di passare dal modo a colori di default ad un modo a colori più veloce, ma con una visualizzazione più a scatti, e ad un modo monocromatico, ancora più veloce. I tre modi vengono selezionati, rispettivamente, premendo i **tasti funzione F3**, **F2** e **F1**.

Gli altri tasti funzione hanno anch'essi un significato; in particolare **F4** consente di modificare il colore dello sfondo (**Paper**) in modo monocromatico mentre **F5** modifica il colore delle scritte (**Ink**) sempre in modo monocromatico.

La combinazione di **tasti Ctrl-F10** esegue un reset dello Spectrum, analogamente all'**istruzione New**.

In definitiva il grado di emulazione dello Spectrum a livello software è pressoché totale e la qualità dell'emulazione è ottima. Vi sono tuttavia alcuni punti deboli, come una maggiore lentezza e pesantezza nell'esecuzione dei programmi, che suggerisce l'espediente di entrare in modo monocromatico quando il colore non sia necessario.

La versione da noi provata ha manifestato il solo difetto di bloccarsi facilmente (non esiste sullo Spectrum niente di pittoresco come un Guru Meditation) se si premono più volte i vari tasti per il cambiamento del modo di visualizzazione. Si tratta in definitiva di un buon prodotto, il cui uso pratico è peraltro molto limitato e destinato prevalentemente ai nostalgici del glorioso ZX Spectrum.

DIGIMAIL Srl
Via Coronelli 10
20146 Milano

SUGGERIMENTI E TRUCCHI VARI

Avventure e role playing games: non capita spesso di trovare trucchi e stratagemmi anche per questi generi di gioco.

In «Chaos Strikes Back» localizzate un drago e lanciate l'incantesimo MON ZO GOR SAR; premete Esc per mettere il gioco in pausa e poi, tenendo premuto il tasto Alt, digitate la frase LORD LIBRASULUS SMITHES THEE DOWN.

Eliminate la pausa, ammazzate il drago e troverete una «firestaff»; a questo punto voi ed il vostro gruppo di avventurieri sarete invincibili.

Ecco invece uno stratagemma utilissimo per riprodurre un oggetto quante volte si desideri, che funziona con «Pool of Radiance», «Curse of the Azure Bonds» e «Champions of Krynn»: create un personaggio nuovo, caricate la situazione salvata e trasferite tutti gli oggetti utili sul nuovo personaggio; togliete il nuovo personaggio dal vostro gruppo e ricaricatelo nuovamente subito dopo; trasferite i suoi oggetti agli altri membri del gruppo, rimuovetelo dal gruppo e ricaricatelo subito dopo. Egli avrà con sé ancora tutti gli oggetti, che potrete trasferire nuovamente agli altri membri per ripetere l'operazione a volontà.

Sul fascicolo di AmigaByte di gennaio abbiamo pubblicato un elenco di codici esadecimali da applicare, tramite un disk editor, sui dischetti di alcuni popolari videogiochi per ottenere vite o energia infinite. Replichiamo qui con una nuova serie di codici, ricordandovi che per poterli utilizzare è necessario disporre di un disk editor, come lo «Smart Disk», che consenta di modificare direttamente sui dischetti il contenuto di tracce e settori.

Ancora una volta avvertiamo che queste modifiche sono permanenti e devono assolutamente esere effettuate su copie di backup dei dischetti, mai sugli originali: altrimenti, in caso di errore o disattenzione, potreste danneggiare per sempre irreparabilmente il programma.

La tabella qui riprodotta contiene le sequenze di byte da ricercare sui dischetti e quella da sostituire per ottenere i risultati indicati a fianco dei rispettivi codici.

BYTE DA RICERCARE	BYTE DA SOSTITUIRE	RISULTATO
	PARADROID '90	
66 1A 30 2C 00 08 53 40	60 1A 30 2C 00 08 53 40	Tempo infinito
20 6D 09 D2 30 28 00 06	4E 75 09 D2 30 28 00 06	Energia infinita
	GOLDEN AXE	
53 28 00 2B 6B 2A 2F 08	60 28 00 2B 6B 2A 2F 08	Vite infinite
4A 28 00 2C 6B 06 4A 28	11 7C 00 03 00 2C 4A 28	Energia infinta
	U.N. SQUADRON	
53 68 00 7A 4E 75 30 BC	4A 68 00 7A 4E 75 30 BC	Crediti infiniti
53 68 00 1E 66 00 00 2E	4A 68 00 1E 66 00 00 2E	Energia infinita
53 68 00 1E 4C DF 7F FF	4A 68 00 1E 4C DF 7F FF	" "
53 68 00 24 4C DF 7F FF	4A 68 00 24 4C DF 7F FF	" "
53 68 00 28 53 40 E5 46	4A 68 00 28 53 40 E5 46	Munizioni infinite
	SPINDIZZY WORLDS	
51 6D 36 0E 64 00 00 3A	60 3E 36 0E 64 00 00 3A	Tempo infinito
61 00 F9 AE 61 00 4E 72	4E 71 4E 71 61 00 4E 72	Energia infinita

Vecchi, ma pur sempre buoni: una serie di trucchi per giochi non più tanto recenti.

Cominciamo dal vecchio «Zany Golf» della Electronic Arts, spiegando come accedere ad un livello segreto di gioco.

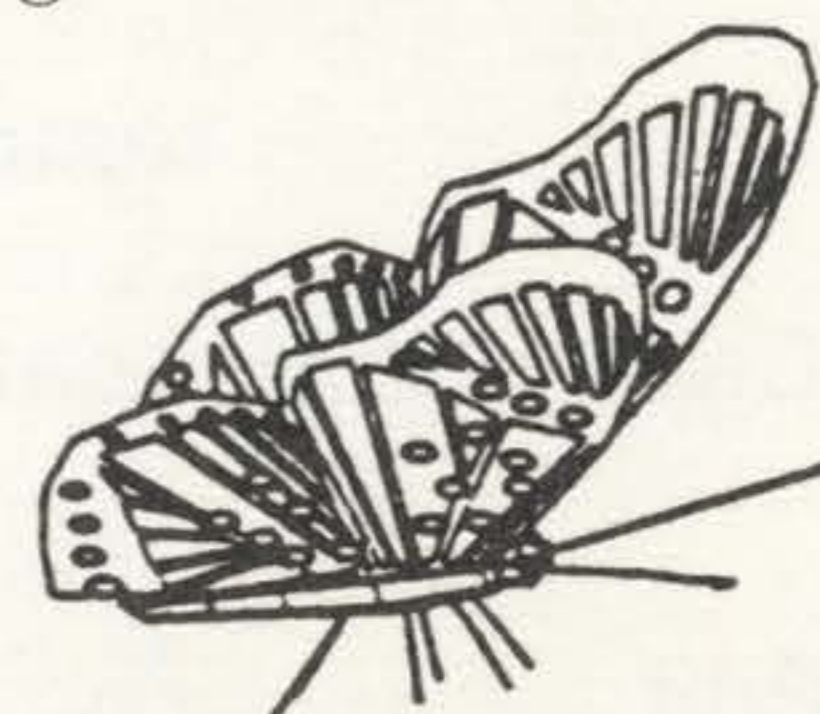
Quando arrivate all'ultima buca di tutto il percorso, tenete d'occhio il foro della tana del topo. Quando vedete apparire gli occhi nel foro, mandateci dentro la palla e verrete trasportati al Mystery Hole.

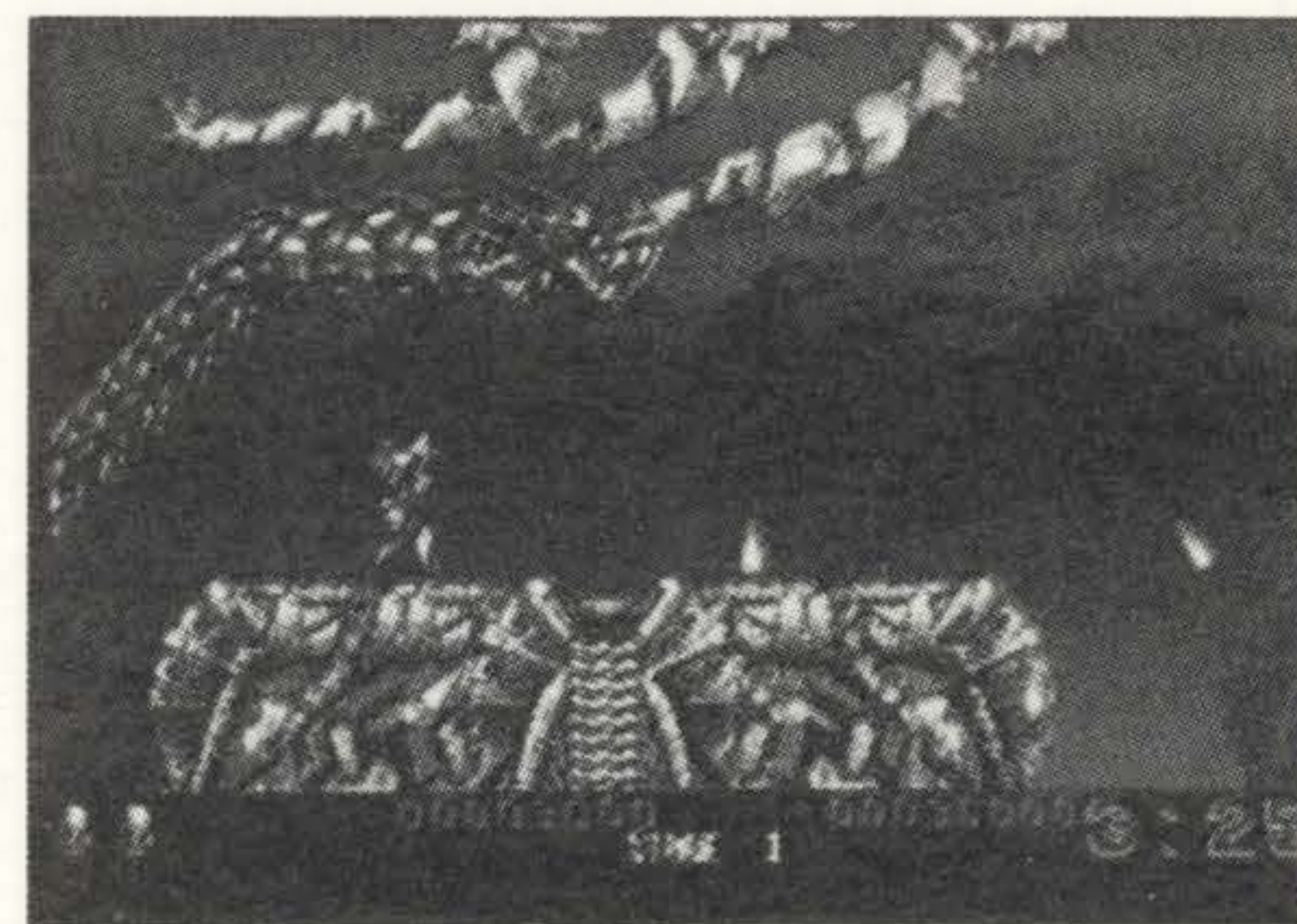
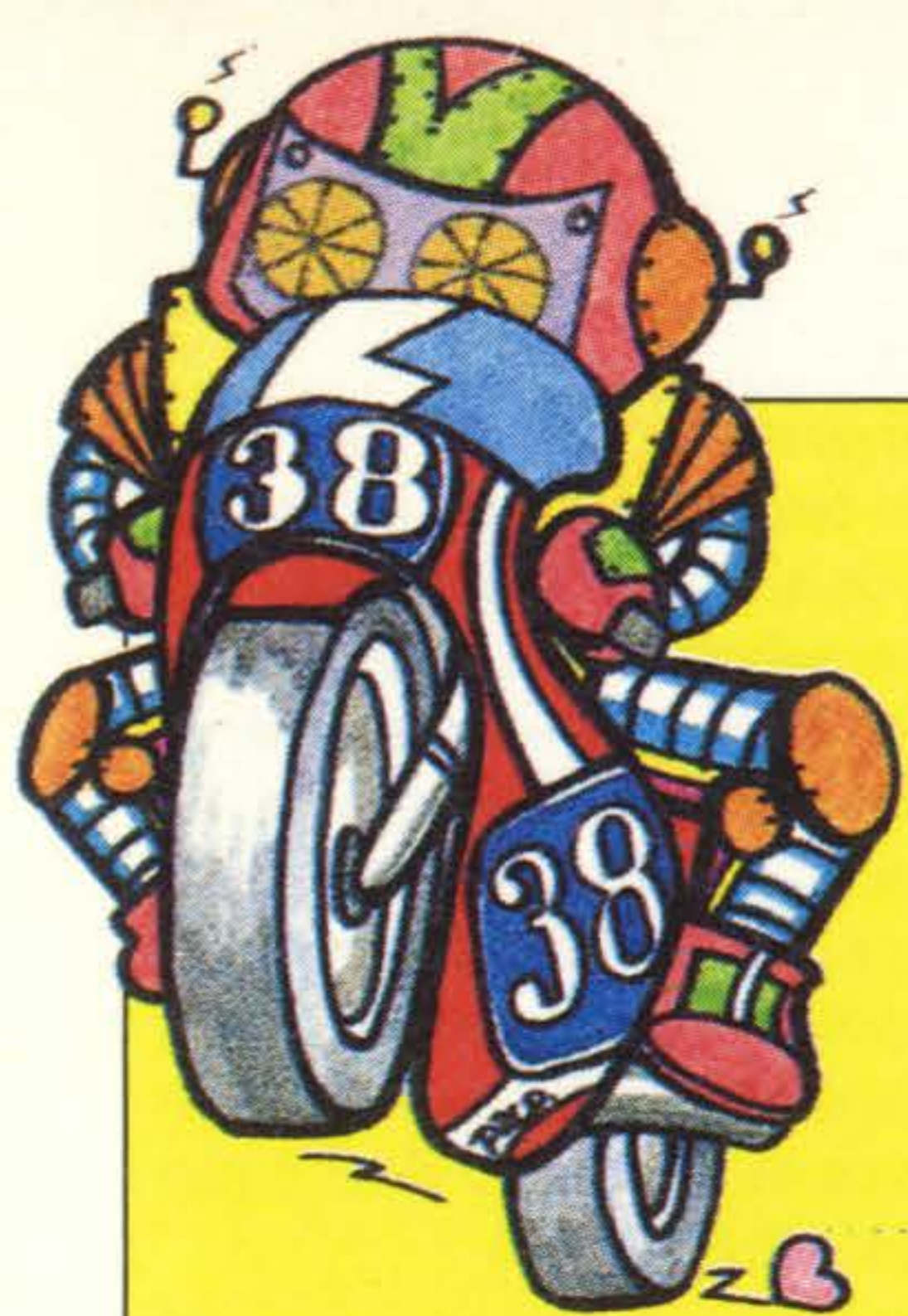
Proseguiamo con «StormTropper»: digitate JAMES CAMERON nella tabella degli hi-scores. Durante il gioco potrete premere il tasto F9 per aggiungere altre vite.

Nella conversione Activision di «Afterburner», digitate la frase TOGETHER IN ELECTRIC DREAMS per attivare i seguenti tasti:

< = Torna al livello precedente
> = Passa al livello seguente
G = Aggiunge missili extra
N = Aggiunge vite extra

Anche in «RoadBlasters» è possibile dare una cosa simile: digitate la parola LAVILLASTRANGIATO subito dopo il caricamento, per attivare questa serie di tasti speciali.

X = Fa ruotare la vettura
S = Salta al livello seguente
P = Rifornisce il serbatoio
G = Termina il gioco
0 = Elimina tutte le armi
1 = Attiva il cannone
2 = Attiva i missili
3 = Attiva gli scudi
4 = Attiva gli iniettori turbo

Occupiamoci di draghi: il recente «Dragon Breed» dell'Activision è la conversione di un gioco arcade originariamente prodotto dalla Irem: mettete il gioco in pausa, e digitate appunto IREM per ottenere vite infinite. Dopo aver attivato il cheat mode, è inoltre possibile premere N per passare al livello successivo.

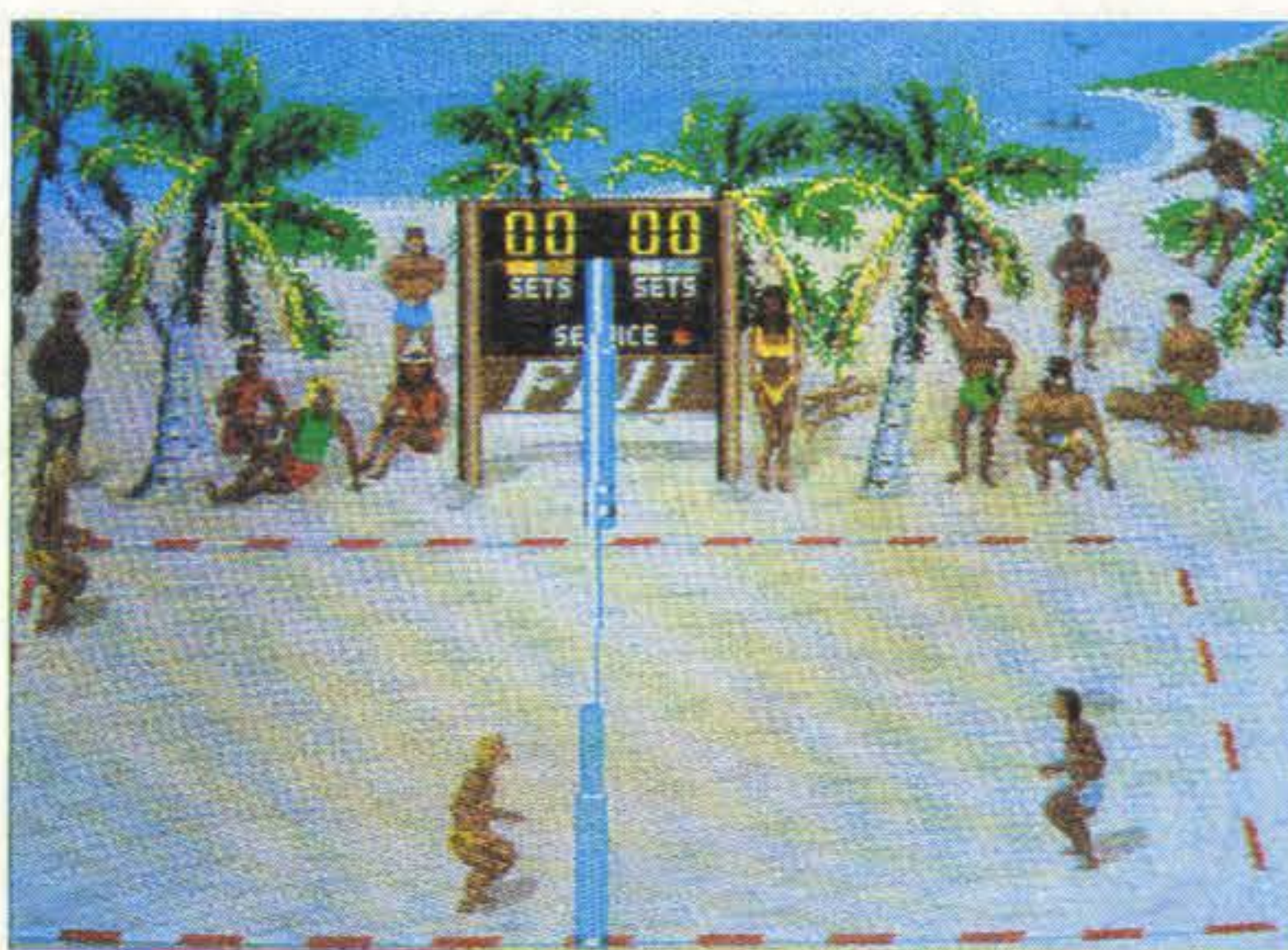
In «Dragon Scape» potete passare al livello successivo premendo in successione il tasto Tab ed il tasto 2.

Infine, in «Dragon Spirit» potete attivare un cheat mode mettendo il gioco in pausa con il tasto F9, digitando le parole DRAGON HEAD e premendo F10.

JUPITER MASTERDRIVE

Sul pianeta Terra, nell'anno 5000, tutti gli sport violenti o pericolosi sono stati vietati e l'unico modo per cimentarsi in una gara automobilistica consiste nel recarsi sul pianeta Giove. Sulle lune di questo pianeta si svolgono nove corse a bordo di veicoli speciali, armabili e potenziabili acquistando accessori prima di ogni gara; il regolamento infatti prevede che per arrivare primi sia tutto lecito, anche sbarazzarsi drasticamente degli avversari.
«Jupiter Masterdrive», della francese

UbiSoft, è molto simile a «Nitro» della Psygnosys ed alla pletora di giochi di corsa automobilistica ripresa dall'alto che sta attualmente riversandosi sul mercato Amiga. Nonostante la scarsa originalità, questo gioco risulta tuttavia molto apprezzabile in quanto molto divertente e giocabile, e tecnicamente molto curato, soprattutto per quanto riguarda il fluidissimo scrolling.
È possibile giocare da soli o contemporaneamente in due: in questo caso,

lo schermo di gioco viene suddiviso verticalmente per visualizzare la sezione di pista in prossimità di ognuno dei concorrenti. La parte inferiore dello schermo contiene, oltre che i vari indicatori di velocità e del tempo, anche un radar indispensabile per localizzare i bonus nascosti disseminati lungo il percorso ed evitare gli ostacoli durante i periodici livelli bonus.
La grafica di «Jupiter Masterdrive» non è certo spettacolare, ma è comunque gradevole ed adeguata al genere di gioco. La giocabilità invece, come abbiamo già detto, è molto elevata ed il divertimento assicurato, specialmente se si gioca in due.

OVER THE NET

Una volta tanto, il «made in Italy» trionfa anche sugli schermi di Amiga: «Over the net» è infatti prodotto dalla software house italiana Genias ed è un'eccellente simulazione arcade sportiva basata sulla variante da spiaggia del gioco della pallavolo noto come «beach volley».
Graficamente gradevole e decisamente divertente, «Over the net» risulta superiore a tutte le versioni a 16 bit della pallavolo che lo hanno finora preceduto, compreso il noto «Beach Volley» della Ocean.
L'azione è ripresa lateralmente, con la rete che divide lo schermo in due e le squadre da entrambi i lati. Ogni squadra è composta da due giocatori, uno dei quali è controllato dal computer in maniera davvero efficiente. A

differenza di tante simulazioni sportive (specialmente quelle calcistiche), il partner computerizzato è molto preciso nel passarvi la palla o nel cogliere i passaggi ed inviarveli. I vari livelli di gioco sono ambientati nelle più famose località balneari del mondo (Ibiza, Miami, le isole Seychelles o, naturalmente, Rimini) anche se non si differenziano granché tra loro dal punto di vista grafico.
L'animazione degli sprite peraltro è molto realistica: i giocatori si spostano con naturalezza, hanno a disposizione una grande quantità di mosse, e rispondono prontamente ai comandi: sebbene non spettacolare, la grafica è nel complesso curata e dettagliata (notate le ombre dei giocatori).
La giocabilità resta comunque il maggiore pregio di «Over the net»: gli amanti della pallavolo non resteranno delusi.

GOLDEN AXE

Fedelissima conversione di un arcade della Sega, «Golden Axe» è tra i giochi di combattimento graficamente più spettacolari disponibili sulla piazza. Il solito malvagio

nemico, questa volta impersonato da un gigante armato di ascia che risponde al nome poco rassicurante di «Death Adder», ha rapito un principe e la sua consorte e li tiene prigionieri nel suo castello. Indovinate a chi tocca andare a salvarli, affrontando le orde di soldati, scheletri, draghi e cavalieri in armatura pronti a tutto pur di farvi a fettine? Chi ve lo fa fare? Innanzitutto, la grafica e la qualità dell'animazione degli sprite, resi con molto realismo; in secondo luogo, la fedeltà di questa conversione rispetto all'originale da bar, che farà felici coloro che hanno sprecato centinaia di monetine tentando di portarlo a termine. Infine, il divertimento di poter giocare contemporaneamente i due e ridurre a carne trita in compagnia di un amico tutti gli odiosi soldati dell'esercito di «Death Adder».
Potete scegliere di quale personaggio rivestire i panni: un barbaro, uno gnomo, o una guerriera. Ognuno dei tre possiede un'arma segreta diversa, attivabile premendo il tasto «Alt», la cui potenza aumenta in relazione al numero di pozioni raccolte lungo la strada. Attenti a non sprecare inutilmente questi bonus perché i nemici di fine livello sono ossi decisamente duri da

Software Express

sconfiggere solo a colpi di spada.
«Golden Axe» non è certo un gioco originale o innovativo, ma risulta ugualmente godibile grazie alla grafica piacevole e, una volta tanto, alla buona giocabilità.

LINE OF FIRE

L'idea di base di «Line of Fire», ennesima conversione di un videogame da bar della Sega, è molto simile a quella del celebre «Operation Thunderbolt»: uno o due giocatori, che in questo caso combattono contemporaneamente, devono farsi largo a colpi di mitra, tentando la fuga dopo aver sottratto una micidiale arma segreta in territorio nemico.
L'azione è ripresa dal punto di vista del

giocatore, con un mirino sullo schermo per aprire il fuoco contro le orde di soldati, velivoli e mezzi corazzati che gli sbarrano la strada. La differenza, rispetto ad «Operation Thunderbolt» risiede nel fatto che qui è possibile non soltanto spostarsi orizzontalmente e verticalmente lungo il percorso, ma anche girare su se stessi, e definire quasi autonomamente la strada da seguire. I livelli di gioco sono otto, al termine di ognuno dei quali attendono l'immancabile super tank o il super-aeroplano, durissimi a morire, che devono essere disintegrati preferibilmente a suon di bombe a mano. Naturalmente le munizioni a disposizione non sono illimitate, ed i rifornimenti devono essere raccolti lungo il percorso.
I livelli e gli scenari sono sufficientemente vari: oltre che negli scontri a fuoco tra soldati nella giungla, nel deserto ed in altri ambienti, capita occasionalmente anche di trovarsi a bordo di elicotteri o motoscafi. La giocabilità di «Line of Fire» è molto elevata, specialmente affrontandolo in due: la grafica ed il sonoro invece lasciano talvolta qualcosa a desiderare, specialmente sotto il punto di vista della definizione degli sprite, che spesso appaiono abbozzati un po' troppo rozzamente. Si tratta comunque di un gioco di ottimo livello, tra i migliori del suo genere.

NARC

Con «Narc» ci avventuriamo nuovamente nel territorio delle conversioni arcade. Prodotto

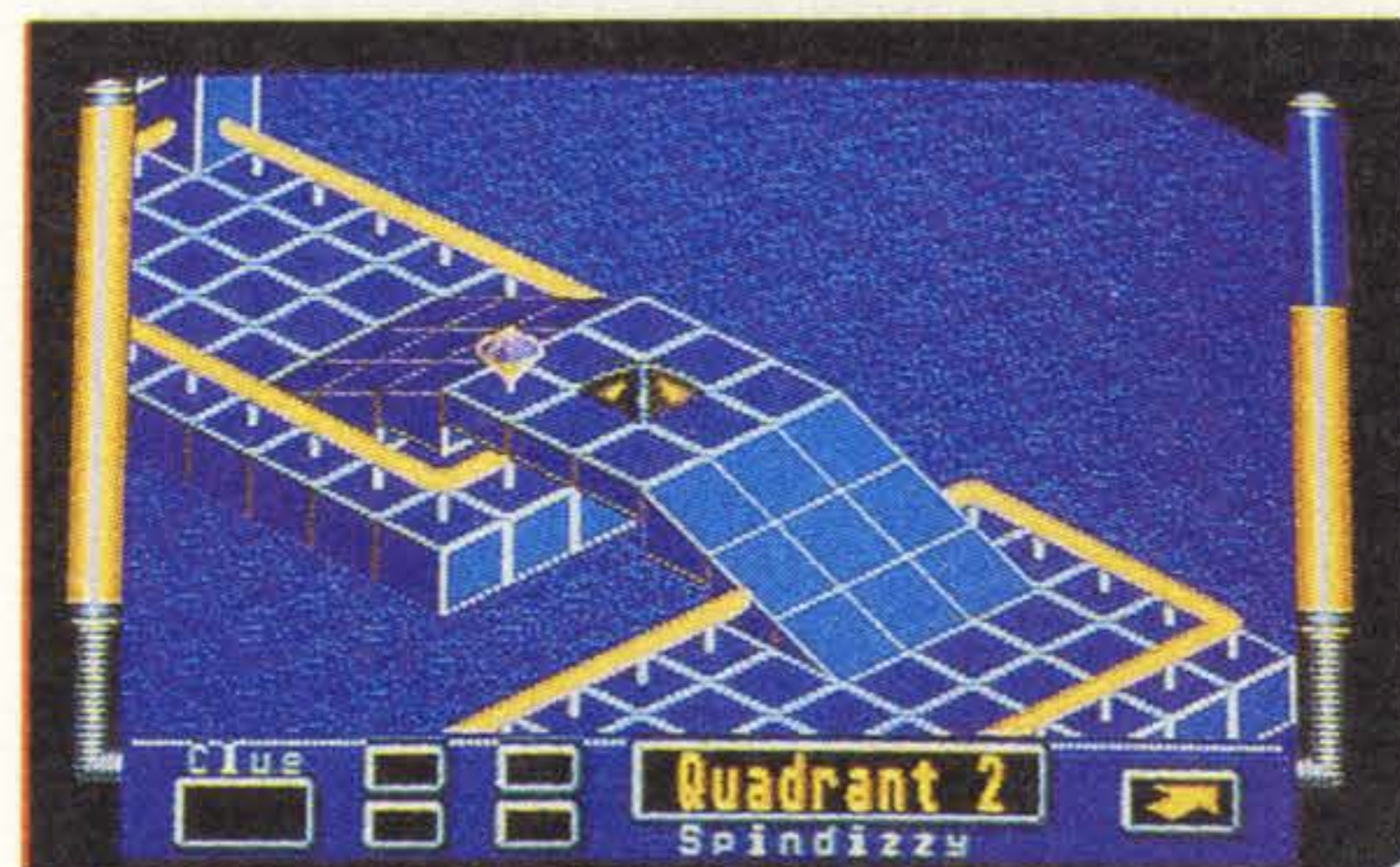
dalla Ocean e basato su di un coin-op targato Williams, «Narc» è ambientato nel mondo della droga del futuro. Uno o due giocatori rivestono il ruolo di poliziotti della squadra narcotici incaricati di penetrare nel territorio occupato da una banda di trafficanti di stupefacenti e di farne letteralmente piazza pulita a colpi di Uzi o di bazooka.
Naturalmente è possibile evitare di polverizzare i malfattori arrestandoli regolarmente, a patto di riuscire ad avvicinarli abbastanza.
Le maniere forti tuttavia solo le più efficaci e vantaggiose: i criminali fatti a brandelli lasciano dietro di sé droga, denaro, munizioni e razzi, che possono essere raccolti per incrementare il proprio punteggio ed armamento.
Oltre che i trafficanti ed i loro sicari, i nostri sbrigativi poliziotti devono fronteggiare un discreto assortimento di nemici, che comprendono cani randagi, drogati che lanciano siringhe, robot da laboratorio e persino clown armati!
«Narc» è molto giocabile e divertente, anche se il buon gusto non è certo la sua dote principale: già la versione arcade era nota per essere piuttosto violenta, ed anche questa conversione non è da meno (osservate gli arti che volano quando usate il bazooka!). Grande divertimento per tutta la famiglia!
La grafica di «Narc» è molto buona, pur non essendo particolarmente spettacolare, e ricorda molto quella di «Ninja Warriors»: la cosa si spiega facilmente, visto che entrambi i giochi sono stati programmati dallo stesso team, The Sales Curve.

PUZZNIC

Questa conversione Ocean di un game da bar della Taito appartiene alla categoria dei

giochi semplici da imparare, difficili da portare a termine, ed impossibili da abbandonare. Il meccanismo è talmente immediato che il manualetto di istruzioni consta, praticamente, di sole poche righe. Sullo schermo appare, in ogni livello, un certo numero di tessere contrassegnate da un simbolo colorato: occorre liberare ogni livello da tutte le tessere in esso contenute prima che scada il tempo. Per far scomparire le tessere, è sufficiente accostarne due o più dello stesso tipo verticalmente od orizzontalmente.
I blocchi possono essere spostati solo in senso orizzontale: a muoverli verticalmente ci pensa la forza di gravità, poiché possono solo cadere verso il basso, senza possibilità di farli risalire.
In caso di errore, ad esempio blocchi posizionati in punti sbagliati e non più recuperabili o singoli blocchi avanzati non più accoppiabili, è possibile ripetere lo stesso livello dal principio premendo la barra spaziatrice: il tempo a disposizione, tuttavia, non riprenderà dall'inizio. Come se non bastasse, con l'avanzare dei livelli aumentano anche il grado di complessità degli schermi ed il numero di ostacoli semoventi.
«Puzznic» non è graficamente eclatante: sotto il profilo estetico, non si distacca molto dallo standard di molti game di pubblico dominio. Sono la giocabilità e la semplicità del suo meccanismo a farne un piccolo gioiello, che dovrebbe appassionare sia i patiti dei rompicapi alla «Tetris» che gli amanti dei giochi arcade più frenetici.

SPINDIZZY WORLDS

Originariamente apparso sugli schermi ad otto bit dell'Amstrad CPC e del Commodore 64 alcuni anni fa, «Spindizzy» fa ora la propria trionfale comparsa anche su Amiga. Ispirato al classico «Marble Madness», vede come protagonista una trottola perennemente in rotazione che risponde al

nome di Gerald. Lo scopo di Gerald consiste nell'esplorare 185 schermi pieni di piattaforme, trappole semoventi ed altri pericoli, raccogliendo tutti i diamanti disseminati lungo il percorso. A complicare le cose contribuisce un limite di tempo: ogni volta che Gerald cade da altezze troppo elevate o viene a contatto con un nemico o con un trabocchetto, i secondi a sua disposizione diminuiscono sempre di più.
La giocabilità è molto elevata, grazie alla varietà di livelli e di mondi da esplorare, e grazie alla relativa facilità con la quale si impara presto a controllare i movimenti di Gerald; il gioco comprende infatti anche un «tutorial» interno di grande aiuto per apprendere i comandi ed i trucchi relativi

agli spostamenti. La grafica è gradevole, anche se non particolarmente spettacolare: per loro stessa natura, le spigolose piattaforme dei mondi di «Spindizzy» non si prestano granché ad effetti grafici troppo suggestivi. Nonostante le apparenze, «Spindizzy» è più simile ad un rompicapo che ad un gioco arcade: per risolvere tutti i problemi che bloccano i progressi di Gerald sono più utili il ragionamento e la strategia che la prontezza di riflessi e l'abilità con il joystick. Per questo «Spindizzy» è l'ideale per chi adora i giochi impegnativi che non si portano a termine in poche ore.

WRAITH OF THE DEMON

La ReadySoft, allontanandosi temporaneamente dallo stile della serie di «Dragon's Lair», ha prodotto con «Wraith of the Demon» uno dei giochi arcade di ambientazione fantasy più riusciti e piacevoli

che si siano visti negli ultimi mesi. L'ispirazione è venuta chiaramente da «Shadow of the Beast»: ma «Wraith of the Demon», pur non eccellendo in originalità, riesce comunque ad eguagliare il celebre gioco della Psygnosys sotto il profilo della grafica e a superarlo sotto quello della giocabilità.
La vostra missione, preceduta dall'immancabile sequenza introduttiva animata, consiste nell'attraversare quindici livelli abitati da creature demoniache sempre diverse e sempre più ardue da eliminare. Al termine del tragitto vi attende il confronto finale con il demone, che tiene prigioniera una bella fata allo scopo di impedirle di

riportare l'ordine e la pace sulla Terra. Alcuni livelli sono a scrolling orizzontale mentre altri sono costituiti da singole schermate: tutti comunque sono graficamente spettacolari e piuttosto ardui da completare.
Lungo la strada potete raccogliere ed impiegare tre diversi tipi di pozione magica: il primo vi garantisce la temporanea invulnerabilità, il secondo consente di distruggere i nemici o di infliggere loro notevoli danni, il terzo vi fa recuperare preziosi punti di energia. Le animazioni degli sprite di «Wraith of the Demon» sono davvero sensazionali, a partire dalla vostra cavalcatura nel primo livello, passando attraverso il buffissimo drago blu del secondo, fino ai capelluti guardiani della palude, agli uomini scorpione e al demone finale.

ANDRÉ PANZA'S KICK BOXING

La boxe è uno sport piuttosto violento, nonostante le limitazioni che il regolamento impone ai pugili durante il combattimento; il «kick boxing» (o «savate»), praticato in alcuni paesi orientali e recentemente diffusosi anche in Europa, ne rappresenta una variante ancora più brutale, in cui ai combattenti è permesso anche tirare calci e colpi simili a quelli del karatè.
La US Gold ha prodotto un gioco arcade

basato su questo sport che prende il nome dal francese André Panza, attuale campione del mondo. Graficamente molto bello e realistico, il gioco presenta un solo difetto, che consiste nella lentezza dei movimenti nelle animazioni e nella scarsa responsività dei pugili ai comandi impartiti loro con il joystick. Il game è comunque molto vario e divertente: oltre che le sequenze di allenamento, necessarie per aumentare i livelli di forza, resistenza e prontezza di riflessi del proprio pugile, «Kick Boxing» prevede nove combattimenti sul ring molto spettacolari con altrettanti avversari, per la conquista del titolo di campione.
Le possibili mosse di ogni pugile sono oltre una cinquantina, ma soltanto una dozzina è attivabile durante un combattimento (anche a causa delle limitazioni imposte dal joystick). Il giocatore può comunque, durante gli allenamenti, selezionare quali mosse adottare per il match e sviluppare quindi uno stile personalizzato a seconda delle esigenze. Tutto sommato «André Panza's Kick Boxing» risulta essere un game di ottimo livello, la cui giocabilità è limitata solo da un'occasionale lentezza eccessiva.